Ph.D. Dissertation

Vocal Accommodation in Human-Computer Interaction: Modeling and Integration into Spoken Dialogue Systems

Eran Raveh

Universität des Saarlandes

Submitted in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy in
Language Science and Technology

Supervisors: Prof. Dr. Bernd Möbius
Dr. Ingmar Steiner

Reviewers: Prof. Dr. Bernd Möbius
Prof. Dr.-Ing. Sebastian Möller

Vocal Accommodation in Human-Computer Interaction: Modeling and Integration into Spoken Dialogue Systems

Dissertation
zur Erlangung des akademischen Grades eines
Doktors der Philosophie
der Philosophischen Fakultät
der Universität des Saarlandes

vorgelegt von

Eran Raveh

aus Haifa, Israel

Saarbrücken, November 2021

Der Dekan	Professor Dr. Augustin Speyer
Berichterstatter	Professor Dr. Bernd Möbius
	Professor Dr.-Ing. Sebastian Möller

Tag der letzten Prüfungsleistung: 30. April 2021

To my family
and all who taught and supported me

Acknowledgments

First and foremost, I would like to thank **Dr. Ingmar Steiner** – for believing in me and giving me this non-trivial chance; for guiding me through the less known aspects of research work; and for teaching and demonstrating high standards of work ethic and the means to achieve it.

Many thanks to **Prof. Dr. Bernd Möbius**, for accompanying me through the convoluted paths of academia, and giving his support all the way to the finish line.

I would like to express my appreciation to **Prof. Dr.-Ing. Sebastian Möller** for his help and feedback that were always quick, precise, and friendly.

To **Iona Gessinger**, who walked this path together with me from start to finish. We made it!

Big thanks to **Dr. Alexander Hewer** and **Dr. Sébastien Le Maguer** for their punctilious feedback and invaluable proofreading.

My sincere thanks and gratitude to **Prof. Dr. Christopher Culy** for believing in my capabilities early on and helping me to pursue further opportunities, and for caring about me and my work also after our period working together.

I would like to thank **Dr. Peter Cahill** and **Dr.-Ing. Christian Dittmar** for exposing me to the vast world of speech processing and giving me the opportunities to be part of their work and learn from their extensive knowledge and experience.

To **Prof. Dr. Detmar Meurers**, who introduced me to the field of computational linguistics in a fascinating and supportive manner.

Finally, to all fellow Ph.D. students, fellow event organizers, colleagues, collaborators, and interesting people I met in conference and workshops around the world. It would not have been the same without you.

Hiermit versichere ich, dass ich die vorgelegte Arbeit selbstständig und nur mit den angegebenen Quellen und Hilfsmitteln einschließlich des WWW und anderer elektronischer Quellen angefertigt habe. Alle Stellen der Arbeit, die ich anderen Werken dem Wortlaut oder dem Sinne nach entnommen habe, sind kenntlich gemacht.

Eran Raveh

Summary

With the rapidly increasing usage of voice-activated devices worldwide, verbal communication with computers is steadily becoming more common. Although speech is the principal natural manner of human communication, it is still challenging for computers, and users had been growing accustomed to adjusting their speaking style for computers. Such adjustments occur naturally, and typically unconsciously, in humans during an exchange to control the social distance between the interlocutors and improve the conversation's efficiency. This phenomenon is called accommodation and it occurs on various modalities in human communication, like hand gestures, facial expressions, eye gaze, lexical and grammatical choices, and others. Vocal accommodation deals with phonetic-level changes occurring in segmental and suprasegmental features. A decrease in the difference between the speakers' feature realizations results in convergence, while an increasing distance leads to divergence. The lack of such mutual adjustments made naturally by humans in computers' speech creates a gap between human-human and human-computer interactions. Moreover, voice-activated systems currently speak in exactly the same manner to all users, regardless of their speech characteristics or realizations of specific features. Detecting phonetic variations and generating adaptive speech output would enhance user personalization, offer more human-like communication, and ultimately should improve the overall interaction experience. Thus, investigating these aspects of accommodation will help to understand and improving human-computer interaction.

This thesis provides a comprehensive overview of the required building blocks for a roadmap toward the integration of accommodation capabilities into spoken dialogue systems. These include conducting human-human and human-computer interaction experiments to examine the differences in vocal behaviors, approaches for modeling these empirical findings, methods for introducing phonetic variations in synthesized speech, and a way to combine all these components into an accommodative system. While each component is a wide research field by itself, they depend on each other and hence should be jointly considered. The overarching goal of this thesis is therefore not only to show how each of the aspects can be further developed, but also to demonstrate and motivate the connections between them. A special emphasis is put throughout the thesis on the

importance of the temporal aspect of accommodation. Humans constantly change their speech over the course of a conversation. Therefore, accommodation processes should be treated as continuous, dynamic phenomena. Measuring differences in a few discrete points, e.g., beginning and end of an interaction, may leave many accommodation events undiscovered or overly smoothed.

To justify the effort of introducing accommodation in computers, it should first be proven that humans even show any phonetic adjustments when talking to a computer as they do with a human being. As there is no definitive metric for measuring accommodation and evaluating its quality, it is important to empirically study humans productions to later use as references for possible behaviors. In this work, this investigation encapsulates different experimental configurations to achieve a better picture of accommodation effects. First, vocal accommodation was inspected where it naturally occurs, namely in spontaneous human-human conversations. For this purpose, a collection of real-world sales conversations, each with a different representative-prospect pair, was collected and analyzed. These conversations offer a glance into accommodation effects in authentic, unscripted interactions with the common goal of negotiating a deal on the one hand, but with the individual facet of each side of trying to get the best terms on the other hand. The conversations were analyzed using cross-correlation and time series techniques to capture the change dynamics over time. It was found that successful conversations are distinguishable from failed ones by multiple measures. Furthermore, the sales representative proved to be better at leading the vocal changes, i.e., making the prospect follow their speech styles rather than the other way around. They also showed a stronger tendency to take that lead at an earlier stage, all the more so in successful conversations. The fact that accommodation occurs more by trained speakers and improves their performances fits anecdotal best practices of sales experts, which are now also proven scientifically. Following these results, the next experiment came closer to the final goal of this work and investigated vocal accommodation effects in human-computer interaction. This was done via a shadowing experiment, which offers a controlled setting for examining phonetic variations. As spoken dialogue systems with such accommodation capabilities (like this work aims to achieve) do not exist yet, a simulated system was used to introduce these changes to the participants, who believed they help with the testing of a language learning tutoring system. After determining their preference concerning three segmental phonetic features, participants were listen-

ing to either natural or synthesized voices of male and female speakers, which produced the participants' dispreferred variation of the aforementioned features. Accommodation occurred in all cases, but the natural voices triggered stronger effects. Nevertheless, it can be concluded that participants were accommodating toward synthetic voices as well, which means that social mechanisms are applied in humans also when speaking with computer-based interlocutors. The shadowing paradigm was utilized also to test whether accommodation is a phenomenon associated only with speech or with other vocal productions as well. To that end, accommodation in the singing of familiar and novel music was examined. Interestingly, accommodation was found in both cases, though in different ways. While participants seemed to use the familiar piece merely as a reference for singing more accurately, the novel piece became the goal for complete replicate. For example, one difference was that mostly pitch corrections were introduced in the former case, while in the latter also key and rhythmic patterns were adopted. Some of those findings were expected and they show that people's more salient features are also harder to modify using external auditory influence. Lastly, a multiparty experiment with spontaneous human-human-computer interactions was carried out to compare accommodation in human-directed and computer-directed speech. The participants solved tasks for which they needed to talk both with a confederate and with an agent. This allows a direct comparison of their speech based on the addressee within the same conversation, which has not been done so far. Results show that some participants' vocal behavior changed similarly when talking to the confederate and the agent, while others' speech varied only with the confederate. Further analysis found that the greatest factor for this difference was the order in which the participants talked with the interlocutors. Apparently, those who first talked to the agent alone saw it more as a social actor in the conversation, while those who interacted with it after talking to the confederate treated it more as a means to achieve a goal, and thus behaved differently with it. In the latter case, the variations in the human-directed speech were much more prominent. Differences were also found between the analyzed features, but the task type did not influence the degree of accommodation effects. The results of these experiments lead to the conclusion that vocal accommodation does occur in human-computer interactions, even if often to lesser degrees.

With the question of whether people accommodate to computer-based interlocutors as well answered, the next step would be to describe accommodative behaviors in a

computer-processable manner. Two approaches are proposed here: computational and statistical. The computational model aims to capture the presumed cognitive process associated with accommodation in humans. This comprises various steps, such as detecting the variable feature's sound, adding instances of it to the feature's mental memory, and determining how much the sound will change while taking into account both its current representation and the external input. Due to its sequential nature, this model was implemented as a pipeline. Each of the pipeline's five steps corresponds to a specific part of the cognitive process and can have one or more parameters to control its output (e.g., the size of the feature's memory or the accommodation pace). Using these parameters, precise accommodative behaviors can be crafted while applying expert knowledge to motivate the chosen parameter values. These advantages make this approach suitable for experimentation with pre-defined, deterministic behaviors where each step can be changed individually. Ultimately, this approach makes a system vocally responsive to users' speech input. The second approach grants more evolved behaviors, by defining different core behaviors and adding non-deterministic variations on top of them. This resembles human behavioral patterns, as each person has a base way of accommodating (or not accommodating), which may arbitrarily change based on the specific circumstances. This approach offers a data-driven statistical way to extract accommodation behaviors from a given collection of interactions. First, the target feature's values of each speaker in an interaction are converted into continuous interpolated lines by drawing one sample from the posterior distribution of a Gaussian process conditioned on the given values. Then, the gradients of these lines, which represent rates of mutual change, are used to defined discrete levels of change based on their distribution. Finally, each level is assigned a symbol, which ultimately creates a symbol sequence representation for each interaction. The sequences are clustered so that each cluster stands for a type of behavior. The sequences of a cluster can then be used to calculate n-gram probabilities that enable the generation of new sequences of the captured behavior. The specific output value is sampled from the range corresponding to the generated symbol. With this approach, accommodation behaviors are extracted directly from data, as opposed to manually crafting them. However, it is harder to describe what exactly these behaviors represent and motivate the use of one of them over the other. To bridge this gap between these two approaches, it is also discussed how they can be combined to benefit from the advantages of both. Furthermore, to generate more structured behaviors, a

hierarchy of accommodation complexity levels is suggested here, from a direct adoption of users' realizations, via specified responsiveness, and up to independent core behaviors with non-deterministic variational productions.

Besides a way to track and represent vocal changes, an accommodative system also needs a text-to-speech component that is able to realize those changes in the system's speech output. Speech synthesis models are typically trained once on data with certain characteristics and do not change afterward. This prevents such models from introducing any variation in specific sounds and other phonetic features. Two methods for directly modifying such features are explored here. The first is based on signal modifications applied to the output signal after it was generated by the system. The processing is done between the timestamps of the target features and uses pre-defined scripts that modify the signal to achieve the desired values. This method is more suitable for continuous features like vowel quality, especially in the case of subtle changes that do not necessarily lead to a categorical sound change. The second method aims to capture phonetic variations in the training data. To that end, a training corpus with phonemic representations is used, as opposed to the regular graphemic representations. This way, the model can learn more direct relations between phonemes and sound instead of surface forms and sound, which, depending on the language, might be more complex and depend on their surrounding letters. The target variations themselves don't necessarily need to be explicitly present in the training data, all time the different sounds are naturally distinguishable. In generation time, the current target feature's state determines the phoneme to use for generating the desired sound. This method is suitable for categorical changes, especially for contrasts that naturally exist in the language. While both methods have certain limitations, they provide a proof of concept for the idea that spoken dialogue systems may phonetically adapt their speech output in real-time and without re-training their text-to-speech models.

To combine the behavior definitions and the speech manipulations, a system is required, which can connect these elements to create a complete accommodation capability. The architecture suggested here extends the standard spoken dialogue system with an additional module, which receives the transcribed speech signal from the speech recognition component without influencing the input to the language understanding component. While language the understanding component uses only textual transcription to determine the user's intention, the added component process the raw signal along with its

phonetic transcription. In this extended architecture, the accommodation model is activated in the added module and the information required for speech manipulation is sent to the text-to-speech component. However, the text-to-speech component now has two inputs, viz. the content of the system's response coming from the language generation component and the states of the defined target features from the added component. An implementation of a web-based system with this architecture is introduced here, and its functionality is showcased by demonstrating how it can be used to conduct a shadowing experiment automatically. This has two main advantage: First, since the system recognizes the participants' phonetic variations and automatically selects the appropriate variation to use in its response, the experimenter saves time and prevents manual annotation errors. The experimenter also automatically gains additional information, like exact timestamps of utterances, real-time visualization of the interlocutors' productions, and the possibility to replay and analyze the interaction after the experiment is finished. The second advantage is scalability. Multiple instances of the system can run on a server and be accessed by multiple clients at the same time. This not only saves time and the logistics of bringing participants into a lab, but also allows running the experiment with different configurations (e.g., other parameter values or target features) in a controlled and reproducible way.

This completes a full cycle from examining human behaviors to integrating accommodation capabilities. Though each part of it can undoubtedly be further investigated, the emphasis here is on how they depend and connect to each other. Measuring changes features without showing how they can be modeled or achieving flexible speech synthesis without considering the desired final output might not lead to the final goal of introducing accommodation capabilities into computers. Treating accommodation in human-computer interaction as one large process rather than isolated sub-problems lays the ground for more comprehensive and complete solutions in the future.

Zusammenfassung (Deutsch)

Heutzutage wird die verbale Interaktion mit Computern immer gebräuchlicher, was der rasant wachsenden Anzahl von sprachaktivierten Geräten weltweit geschuldet ist. Allerdings stellt die computerseitige Handhabung gesprochener Sprache weiterhin eine große Herausforderung dar, obwohl sie die bevorzugte Art zwischenmenschlicher Kommunikation repräsentiert. Dieser Umstand führt auch dazu, dass Benutzer ihren Sprachstil an das jeweilige Gerät anpassen, um diese Handhabung zu erleichtern. Solche Anpassungen kommen in menschlicher gesprochener Sprache auch in der zwischenmenschlichen Kommunikation vor. Üblicherweise ereignen sie sich unbewusst und auf natürliche Weise während eines Gesprächs, etwa um die soziale Distanz zwischen den Gesprächsteilnehmern zu kontrollieren oder um die Effizienz des Gesprächs zu verbessern. Dieses Phänomen wird als Akkommodation bezeichnet und findet auf verschiedene Weise während menschlicher Kommunikation statt. Sie äußert sich zum Beispiel in der Gestik, Mimik, Blickrichtung oder aber auch in der Wortwahl und dem verwendeten Satzbau. Vokal-Akkommodation beschäftigt sich mit derartigen Anpassungen auf phonetischer Ebene, die sich in segmentalen und suprasegmentalen Merkmalen zeigen. Werden Ausprägungen dieser Merkmale bei den Gesprächsteilnehmern im Laufe des Gesprächs ähnlicher, spricht man von Konvergenz, vergrößern sich allerdings die Unterschiede, so wird dies als Divergenz bezeichnet. Dieser natürliche gegenseitige Anpassungsvorgang fehlt jedoch auf der Seite des Computers, was zu einer Lücke in der Mensch-Maschine-Interaktion führt. Darüber hinaus verwenden sprachaktivierte Systeme immer dieselbe Sprachausgabe und ignorieren folglich etwaige Unterschiede zum Sprachstil des momentanen Benutzers. Die Erkennung dieser phonetischen Abweichungen und die Erstellung von anpassungsfähiger Sprachausgabe würden zur Personalisierung dieser Systeme beitragen und könnten letztendlich die insgesamte Benutzererfahrung verbessern. Aus diesem Grund kann die Erforschung dieser Aspekte von Akkommodation helfen, Mensch-Maschine-Interaktion besser zu verstehen und weiterzuentwickeln.

Die vorliegende Dissertation stellt einen umfassenden Überblick zu Bausteinen bereit, die nötig sind, um Akkommodationsfähigkeiten in Sprachdialogsysteme zu integrieren. In diesem Zusammenhang wurden auch interaktive Mensch-Mensch- und Mensch-Maschine-Experimente durchgeführt. In diesen Experimenten wurden Differenzen der

vokalen Verhaltensweisen untersucht und Methoden erforscht, wie phonetische Abweichungen in synthetische Sprachausgabe integriert werden können. Um die erhaltenen Ergebnisse empirisch auswerten zu können, wurden hierbei auch verschiedene Modellierungsansätze erforscht. Fernerhin wurde der Frage nachgegangen, wie sich die betreffenden Komponenten kombinieren lassen, um ein Akkommodationssystem zu konstruieren. Jeder dieser Aspekte stellt für sich genommen bereits einen überaus breiten Forschungsbereich dar. Allerdings sind sie voneinander abhängig und sollten zusammen betrachtet werden. Aus diesem Grund liegt ein übergreifender Schwerpunkt dieser Dissertation darauf, nicht nur aufzuzeigen, wie sich diese Aspekte weiterentwickeln lassen, sondern auch zu motivieren, wie sie zusammenhängen. Ein weiterer Schwerpunkt dieser Arbeit befasst sich mit der zeitlichen Komponente des Akkommodationsprozesses, was auf der Beobachtung fußt, dass Menschen im Laufe eines Gesprächs ständig ihren Sprachstil ändern. Diese Beobachtung legt nahe, derartige Prozesse als kontinuierliche und dynamische Prozesse anzusehen. Fasst man jedoch diesen Prozess als diskret auf und betrachtet z.B. nur den Beginn und das Ende einer Interaktion, kann dies dazu führen, dass viele Akkommodationsereignisse unentdeckt bleiben oder übermäßig geglättet werden.

Um die Entwicklung eines vokalen Akkommodationssystems zu rechtfertigen, muss zuerst bewiesen werden, dass Menschen bei der vokalen Interaktion mit einem Computer ein ähnliches Anpassungsverhalten zeigen wie bei der Interaktion mit einem Menschen. Da es keine eindeutig festgelegte Metrik für das Messen des Akkommodationsgrades und für die Evaluierung der Akkommodationsqualität gibt, ist es besonders wichtig, die Sprachproduktion von Menschen empirisch zu untersuchen, um sie als Referenz für mögliche Verhaltensweisen anzuwenden. In dieser Arbeit schließt diese Untersuchung verschiedene experimentelle Anordnungen ein, um einen besseren Überblick über Akkommodationseffekte zu erhalten. In einer ersten Studie wurde die vokale Akkommodation in einer Umgebung untersucht, in der sie natürlich vorkommt: in einem spontanen Mensch-Mensch Gespräch. Zu diesem Zweck wurde eine Sammlung von echten Verkaufsgesprächen gesammelt und analysiert, wobei in jedem dieser Gespräche ein anderes Handelsvertreter-Neukunde Paar teilgenommen hatte. Diese Gespräche verschaffen einen Einblick in Akkommodationseffekte während spontanen authentischen Interaktionen, wobei die Gesprächsteilnehmer zwei Ziele verfolgen: zum einen soll ein Geschäft verhandelt werden, zum anderen möchte aber jeder Teilnehmer für sich die besten Bedingungen aushandeln. Die Konversationen wurde durch das Kreuzkorrelation-Zeitreihen-Verfahren

analysiert, um die dynamischen Änderungen im Zeitverlauf zu erfassen. Hierbei kam zum Vorschein, dass sich erfolgreiche Konversationen von fehlgeschlagenen Gesprächen deutlich unterscheiden lassen. Überdies wurde festgestellt, dass die Handelsvertreter die treibende Kraft von vokalen Änderungen sind, d.h. sie können die Neukunden eher dazu zu bringen, ihren Sprachstil anzupassen, als andersherum. Es wurde auch beobachtet, dass sie diese Akkommodation oft schon zu einem frühen Zeitpunkt auslösen, was besonders bei erfolgreichen Gesprächen beobachtet werden konnte. Dass diese Akkommodation stärker bei trainierten Sprechern ausgelöst wird, deckt sich mit den meist anekdotischen Empfehlungen von erfahrenen Handelsvertretern, die bisher nie wissenschaftlich nachgewiesen worden sind. Basierend auf diesen Ergebnissen beschäftigte sich die nächste Studie mehr mit dem Hauptziel dieser Arbeit und untersuchte Akkommodationseffekte bei Mensch-Maschine-Interaktionen. Diese Studie führte ein Shadowing-Experiment durch, das ein kontrolliertes Umfeld für die Untersuchung phonetischer Abweichungen anbietet. Da Sprachdialogsysteme mit solchen Akkommodationsfähigkeiten noch nicht existieren, wurde stattdessen ein simuliertes System eingesetzt, um diese Akkommodationsprozesse bei den Teilnehmern auszulösen, wobei diese im Glauben waren, ein Sprachlernsystem zu testen. Nach der Bestimmung ihrer Präferenzen hinsichtlich dreier segmentaler Merkmale hörten die Teilnehmer entweder natürlichen oder synthetischen Stimmen von männlichen und weiblichen Sprechern zu, die nicht die bevorzugten Variation der oben genannten Merkmale produzierten. Akkommodation fand in allen Fällen statt, obwohl die natürlichen Stimmen stärkere Effekte auslösten. Es kann jedoch gefolgert werden, dass Teilnehmer sich auch an den synthetischen Stimmen orientierten, was bedeutet, dass soziale Mechanismen bei Menschen auch beim Sprechen mit Computern angewendet werden. Das Shadowing-Paradigma wurde auch verwendet, um zu testen, ob Akkommodation ein nur mit Sprache assoziiertes Phänomen ist oder ob sie auch in anderen vokalen Aktivitäten stattfindet. Hierzu wurde Akkommodation im Gesang zu vertrauter und unbekannter Musik untersucht. Interessanterweise wurden in beiden Fällen Akkommodationseffekte gemessen, wenn auch nur auf unterschiedliche Weise. Wohingegen die Teilnehmer das vertraute Stück lediglich als Referenz für einen genaueren Gesang zu verwenden schienen, wurde das neuartige Stück zum Ziel einer vollständigen Nachbildung. Ein Unterschied bestand z.B. darin, dass im ersteren Fall hauptsächlich Tonhöhenkorrekturen durchgeführt wurden, während im zweiten Fall auch Tonart und Rhythmusmuster übernommen wurden. Einige dieser Ergebnisse wurden erwartet und

zeigen, dass die hervorstechenderen Merkmale von Menschen auch durch externen auditorischen Einfluss schwerer zu modifizieren sind. Zuletzt wurde ein Mehrparteienexperiment mit spontanen Mensch-Mensch-Computer-Interaktionen durchgeführt, um Akkommodation in mensch- und computergerichteter Sprache zu vergleichen. Die Teilnehmer lösten Aufgaben, für die sie sowohl mit einem Konföderierten als auch mit einem Agenten sprechen mussten. Dies ermöglicht einen direkten Vergleich ihrer Sprache basierend auf dem Adressaten innerhalb derselben Konversation, was bisher noch nicht erforscht worden ist. Die Ergebnisse zeigen, dass sich das vokale Verhalten einiger Teilnehmer im Gespräch mit dem Konföderierten und dem Agenten ähnlich änderte, während die Sprache anderer Teilnehmer nur mit dem Konföderierten variierte. Weitere Analysen ergaben, dass der größte Faktor für diesen Unterschied die Reihenfolge war, in der die Teilnehmer mit den Gesprächspartnern sprachen. Anscheinend sahen die Teilnehmer, die zuerst mit dem Agenten allein sprachen, ihn eher als einen sozialen Akteur im Gespräch, während diejenigen, die erst mit dem Konföderierten interagierten, ihn eher als Mittel zur Erreichung eines Ziels betrachteten und sich deswegen anders verhielten. Im letzteren Fall waren die Variationen in der menschgerichteten Sprache viel ausgeprägter. Unterschiede wurden auch zwischen den analysierten Merkmalen festgestellt, aber der Aufgabentyp hatte keinen Einfluss auf den Grad der Akkommodationseffekte. Die Ergebnisse dieser Experimente lassen den Schluss zu, dass bei Mensch-Computer-Interaktionen vokale Akkommodation auftritt, wenn auch häufig in geringerem Maße.

Da nun eine Bestätigung dafür vorliegt, dass Menschen auch bei der Interaktion mit Computern ein Akkommodationsverhalten aufzeigen, liegt der Schritt nahe, dieses Verhalten auf eine computergestützte Weise zu beschreiben. Hier werden zwei Ansätze vorgeschlagen: ein Ansatz basierend auf einem Rechenmodell und einer basierend auf einem statistischen Modell. Das Ziel des Rechenmodells ist es, den vermuteten kognitiven Prozess zu erfassen, der mit der Akkommodation beim Menschen verbunden ist. Dies umfasst verschiedene Schritte, z.B. das Erkennen des Klangs des variablen Merkmals, das Hinzufügen von Instanzen davon zum mentalen Gedächtnis des Merkmals und das Bestimmen, wie stark sich das Merkmal ändert, wobei sowohl seine aktuelle Darstellung als auch die externe Eingabe berücksichtigt werden. Aufgrund seiner sequenziellen Natur wurde dieses Modell als eine Pipeline implementiert. Jeder der fünf Schritte der Pipeline entspricht einem bestimmten Teil des kognitiven Prozesses und kann einen oder mehrere Parameter zur Steuerung seiner Ausgabe aufweisen (z.B. die Größe des Ge-

dächtnisses des Merkmals oder die Akkommodationsgeschwindigkeit). Mit Hilfe dieser Parameter können präzise akkommodative Verhaltensweisen zusammen mit Expertenwissen erstellt werden, um die ausgewählten Parameterwerte zu motivieren. Durch diese Vorteile ist diesen Ansatz besonders zum Experimentieren mit vordefinierten, deterministischen Verhaltensweisen geeignet, bei denen jeder Schritt einzeln geändert werden kann. Letztendlich macht dieser Ansatz ein System stimmlich auf die Spracheingabe von Benutzern ansprechbar. Der zweite Ansatz gewährt weiterentwickelte Verhaltensweisen, indem verschiedene Kernverhalten definiert und nicht deterministische Variationen hinzugefügt werden. Dies ähnelt menschlichen Verhaltensmustern, da jede Person eine grundlegende Art von Akkommodationsverhalten hat, das sich je nach den spezifischen Umständen willkürlich ändern kann. Dieser Ansatz bietet eine datengesteuerte statistische Methode, um das Akkommodationsverhalten aus einer bestimmten Sammlung von Interaktionen zu extrahieren. Zunächst werden die Werte des Zielmerkmals jedes Sprechers in einer Interaktion in kontinuierliche interpolierte Linien umgewandelt, indem eine Probe aus der a posteriori Verteilung eines Gaußprozesses gezogen wird, der von den angegebenen Werten abhängig ist. Dann werden die Gradienten dieser Linien, die die gegenseitigen Änderungsraten darstellen, verwendet, um diskrete Änderungsniveaus basierend auf ihren Verteilungen zu definieren. Schließlich wird jeder Ebene ein Symbol zugewiesen, das letztendlich eine Symbolsequenzdarstellung für jede Interaktion darstellt. Die Sequenzen sind geclustert, sodass jeder Cluster für eine Art von Verhalten steht. Die Sequenzen eines Clusters können dann verwendet werden, um N-Gramm Wahrscheinlichkeiten zu berechnen, die die Erzeugung neuer Sequenzen des erfassten Verhaltens ermöglichen. Der spezifische Ausgabewert wird aus dem Bereich abgetastet, der dem erzeugten Symbol entspricht. Bei diesem Ansatz wird das Akkommodationsverhalten direkt aus Daten extrahiert, anstatt manuell erstellt zu werden. Es kann jedoch schwierig sein, zu beschreiben, was genau jedes Verhalten darstellt und die Verwendung eines von ihnen gegenüber dem anderen zu motivieren. Um diesen Spalt zwischen diesen beiden Ansätzen zu schließen, wird auch diskutiert, wie sie kombiniert werden könnten, um von den Vorteilen beider zu profitieren. Darüber hinaus, um strukturiertere Verhaltensweisen zu generieren, wird hier eine Hierarchie von Akkommodationskomplexitätsstufen vorgeschlagen, die von einer direkten Übernahme der Benutzerrealisierungen über eine bestimmte Änderungssensitivität und bis hin zu unabhängigen Kernverhalten mit nicht-deterministischen Variationsproduktionen reicht.

Neben der Möglichkeit, Stimmänderungen zu verfolgen und darzustellen, benötigt ein akkommodatives System auch eine Text-zu-Sprache Komponente, die diese Änderungen in der Sprachausgabe des Systems realisieren kann. Sprachsynthesemodelle werden in der Regel einmal mit Daten mit bestimmten Merkmalen trainiert und ändern sich danach nicht mehr. Dies verhindert, dass solche Modelle Variationen in bestimmten Klängen und anderen phonetischen Merkmalen generieren können. Zwei Methoden zum direkten Ändern solcher Merkmale werden hier untersucht. Die erste basiert auf Signalverarbeitung, die auf das Ausgangssignal angewendet wird, nachdem es vom System erzeugt wurde. Die Verarbeitung erfolgt zwischen den Zeitstempeln der Zielmerkmale und verwendet vordefinierte Skripte, die das Signal modifizieren, um die gewünschten Werte zu erreichen. Diese Methode eignet sich besser für kontinuierliche Merkmale wie Vokalqualität, insbesondere bei subtilen Änderungen, die nicht unbedingt zu einer kategorialen Klangänderung führen. Die zweite Methode zielt darauf ab, phonetische Variationen in den Trainingsdaten zu erfassen. Zu diesem Zweck wird im Gegensatz zu den regulären graphemischen Darstellungen ein Trainingskorpus mit phonemischen Darstellungen verwendet. Auf diese Weise kann das Modell direktere Beziehungen zwischen Phonemen und Klang anstelle von Oberflächenformen und Klang erlernen, die je nach Sprache komplexer und von ihren umgebenden Buchstaben abhängen können. Die Zielvariationen selbst müssen nicht unbedingt explizit in den Trainingsdaten enthalten sein, solange die verschiedenen Klänge natürlich immer unterscheidbar sind. In der Generierungsphase bestimmt der Zustand des aktuellen Zielmerkmals das Phonem, das zum Erzeugen des gewünschten Klangs verwendet werden sollte. Diese Methode eignet sich für kategoriale Änderungen, insbesondere für Kontraste, die sich natürlich in der Sprache unterscheiden. Obwohl beide Methoden eindeutig verschiedene Einschränkungen aufweisen, liefern sie einen Machbarkeitsnachweis für die Idee, dass Sprachdialogsysteme ihre Sprachausgabe in Echtzeit phonetisch anpassen können, ohne ihre Text-zu-Sprache Modelle wieder zu trainieren.

Um die Verhaltensdefinitionen und die Sprachmanipulation zu kombinieren, ist ein System erforderlich, das diese Elemente verbinden kann, um ein vollständiges akkommodationsfähiges System zu schaffen. Die hier vorgeschlagene Architektur erweitert den Standardfluss von Sprachdialogsystemen um ein zusätzliches Modul, das das transkribierte Sprachsignal von der Spracherkennungskomponente empfängt, ohne die Eingabe in die Sprachverständniskomponente zu beeinflussen. Während die Sprachverständnis-

komponente nur die Texttranskription verwendet, um die Absicht des Benutzers zu bestimmen, verarbeitet die hinzugefügte Komponente das Rohsignal zusammen mit seiner phonetischen Transkription. In dieser erweiterten Architektur wird das Akkommodationsmodell in dem hinzugefügten Modul aktiviert und die für die Sprachmanipulation erforderlichen Informationen werden an die Text-zu-Sprache Komponente gesendet. Die Text-zu-Sprache Komponente hat jetzt zwei Eingaben, nämlich den Inhalt der Systemantwort, der von der Sprachgenerierungskomponente stammt, und die Zustände der definierten Zielmerkmale von der hinzugefügten Komponente. Hier wird eine Implementierung eines webbasierten Systems mit dieser Architektur vorgestellt und dessen Funktionalitäten wurden durch ein Vorzeigeszenario demonstriert, indem es verwendet wird, um ein Shadowing-Experiment automatisch durchzuführen. Dies hat zwei Hauptvorteile: Erstens spart der Experimentator Zeit und vermeidet manuelle Annotationsfehler, da das System die phonetischen Variationen der Teilnehmer erkennt und automatisch die geeignete Variation für die Rückmeldung auswählt. Der Experimentator erhält außerdem automatisch zusätzliche Informationen wie genaue Zeitstempel der Äußerungen, Echtzeitvisualisierung der Produktionen der Gesprächspartner und die Möglichkeit, die Interaktion nach Abschluss des Experiments erneut abzuspielen und zu analysieren. Der zweite Vorteil ist Skalierbarkeit. Mehrere Instanzen des Systems können auf einem Server ausgeführt werden, auf die mehrere Clients gleichzeitig zugreifen können. Dies spart nicht nur Zeit und Logistik, um Teilnehmer in ein Labor zu bringen, sondern ermöglicht auch die kontrollierte und reproduzierbare Durchführung von Experimenten mit verschiedenen Konfigurationen (z.B. andere Parameterwerte oder Zielmerkmale).

Dies schließt einen vollständigen Zyklus von der Untersuchung des menschlichen Verhaltens bis zur Integration der Akkommodationsfähigkeiten ab. Obwohl jeder Teil davon zweifellos weiter untersucht werden kann, liegt der Schwerpunkt hier darauf, wie sie voneinander abhängen und sich miteinander kombinieren lassen. Das Messen von Änderungsmerkmalen, ohne zu zeigen, wie sie modelliert werden können, oder das Erreichen einer flexiblen Sprachsynthese ohne Berücksichtigung der gewünschten endgültigen Ausgabe führt möglicherweise nicht zum endgültigen Ziel, Akkommodationsfähigkeiten in Computer zu integrieren. Indem diese Dissertation die Vokal-Akkommodation in der Mensch-Computer-Interaktion als einen einzigen großen Prozess betrachtet und nicht als eine Sammlung isolierter Unterprobleme, schafft sie ein Fundament für umfassendere und vollständigere Lösungen in der Zukunft.

Table of Contents

List of Tables

List of Figures

List of Formulae and Algorithms

Notation

$p(x \mid y,\ z)$	conditional probability of x given the context y, z
$\vec{r}$	vector r
$\mathcal{A}^{m \times n}$	matrix A with dimension m over n
$\mathcal{A}^{\mathbb{R} \times \mathbb{R}}$	matrix A describing a 2-dimensional space of real numbers
$\mathcal{G} : \mathbb{Q}^{n \times m} \longrightarrow \mathbb{Q}^m$	a function $\mathcal{G}$ that maps a $n \times m$ rational numbers matrix to a m-dimensional vector of rational numbers.
$\mathcal{N}$	normal (Gaussian) distribution
$X \sim \mathcal{N}(\mu, \sigma^2)$	random variable X has a normal distribution with mean μ and variance σ^2
$\int_i^j f(x)$	integral over the function $f(x)$ between the points i and j, representing the conceptual area under the curve of the function.
$k\left(\theta, x, x'\right)$	covariance function (kernel) k between all possible input pairs using hyperparameter vector θ. Can be shorted-noted as $\mathcal{K}$. x and x' are two data vectors.
$\Sigma(\vec{x})$	covariance matrix (e.g., of a Gaussian process) given by $\Sigma_{i,j} = k(x_i, x_j)$, where k is a positive definite kernel function
$p\left(f\left(\vec{x}\right)\right) \sim \mathcal{GP}\left(m(\vec{x}), \Sigma(\vec{x})\right)$	probability of values of function f over vector x is specified by a Gaussian process with mean function m and covariance function K
$f(X)$	a vector of function values, whose ith element is given by $f(x_i)$
x_*	a not-yet-observed outcome value; similarly, f_* collectively denotes all non-observed outcome values of a function, and Σ_* the covariance values of non-observed values
$\|\mathbf{p} - \mathbf{q}\|_d$	the Euclidean distance between points p and q in a d-dimensional space
	a note raised by one quarter tone
	a note raised by three quarter tones

List of Acronyms

AE account executive 54, 56 ff., 69, 71, 73

AI artificial intelligence 26, 41

AR articulation rate . . 21, 109 ff., 119 f., 130, 169

ASP additional speech processing . 169, 176, 181

ASR automatic speech recognition 33, 44, 57, 120, 129, 131, 138, 162, 164, 169, 176, 180 f., 190

B2B business-to-business 56–59

BPM beats per minute . . . . . 82 f.

C-AI conversational AI 104

C-IQ/CI conversation intelligence (a.k.a. conversation IQ) 54–57

C&C command and control . . 34, 36, 45

CAPT computer-assisted pronunciation training . . . . . 45, 133 f., 185, 191

CASA Computers Are Social Actors 27, 88, 192

CAT communication accommodation theory 4 f., 12, 14, 16, 18, 23, 27, 46, 59, 128

CRM customer relations management 54

CRQA cross-recurrence quantification analysis 54, 59–66, 71

DDS device-directed speech . 105 f., 108, 111, 115, 120

DiD difference-in-difference . . . . . 24 f.

DL deep learning

DM dialogue manager . 34, 44, 169, 176, 181

$\mathbf{f_0}$ fundamental frequency 21, 39, 48, 54, 59, 78, 91, 110 f., 115, 120, 128, 130, 138, 156, 169

GP Gaussian process . 140 ff., 144, 146, 151, 154

GUI graphical user interface . 90, 173 ff., 177, 182, 229

HCI human-computer interaction 3–7, 11, 15, 19, 21, 23, 26 ff., 40 ff., 47 f., 75, 77, 88, 100 f., 104, 106, 110, 123, 128, 139, 144, 166, 173 ff., 183, 190, 192

HDS human-directed speech 105 f., 108, 111, 115, 120

HHCI human-human-computer interaction 7, 103–106, 111, 120

HHI human-human interaction . . . 4 f., 7, 14, 18 f., 21, 25–28, 38–41, 46, 48, 53, 57, 59 f., 71, 75, 88, 101, 104, 120, 123, 128, 139, 166, 174, 190, 192

HMM hidden Markov model . 91, 95 ff., 100

ITS intelligent tutoring system . 32, 104

LOESS locally estimated scatterplot smoothing 115, 143

LoS line of synchrony . . . . . 61, 66 f.

LTAS long-term average spectrum . . 54

MBROLA Multi-Band Resynthesis Overlap-Add 91

MFCC mel-frequency cepstral coefficient 54, 190

ML machine learning 13, 41, 55

NLG natural language generation . . 35, 43 f., 169, 176, 181

NLP natural language processing 32, 47

NLU natural language understanding 33 ff., 169, 176, 181

PA personal assistant5, 23, 25, 36, 38, 45, 104

PAA piecewise aggregate approximation 147 f., 152, 154

PCA principal component analysis . 151

QT quarter tone 82

RQA recurrence quantification analysis 61, 67

RR recurrence rate . . . . . 65 f., 68

SAX symbolic aggregate approximation 148 f., 154

SDR sales development representative 56

SDS spoken dialogue system . . 5–8, 16, 25, 27 f., 31–36, 38, 40 f., 43 ff., 47 f., 88, 129, 135, 139 f., 154, 161 f., 169, 173–176, 180 f., 190 f.

seq2seq sequence-to-sequence 91

SMO sequential minimization optimization 186

SVM support vector machine 186

TAMA time-aligned moving average 143

TTS text-to-speech . . . . 4 f., 16, 35 f., 43–47, 134, 154, 156, 162, 167, 169 f., 176, 181, 190

VA voice assistant . . . . 104 ff., 120, 123

VACC Voice Assistant Conversation Corpus 106, 108

VH virtual human 39

I

Background

Chapter 1

Introduction

SPOKEN human communication is complex and dynamic. One reason for that is the influence interlocutors have on each other while talking. The rise of voice-activated devices challenges the way people use their voice to communicate. Since computers do not pose the same inert tendency to be influenced by human speech, it is an open question whether human communicate the same with computers and whether computers can simulate these natural dynamic changes. This chapter provides the motivation for studying verbal human-computer interaction and the main goals of this work.

1.1 Motivation and goals

People tend to adopt certain behavioral patterns from one another while interacting. These may range from simple physical postures to language usage and even emotional reactions. The overarching term for this phenomenon is *accommodation* and it is commonly occurring in human-human interaction. As explained by communication accommodation theory from the 1970s, accommodation often has a social motive and it is used, even if unconsciously, to identify oneself with certain addressees or to trigger greater likability among a social group (Giles, 2007). It is theorized that the intrinsic motivation for this mechanism is a decrease (or an increase) of the social distance between interlocutors and the improvement of the interaction's overall efficiency (Gallois and Giles, 2015). Various empirical experiments have found accommodation effects in a variety of modalities, like facial expressions (Kinsbourne and Jordan, 2009), eye gaze (Leong et al., 2017), lexical choices (Brennan, 1996), and more. Changes in speech, and especially low-level phonetic ones, are sometimes more subtle and harder to spot, e.g., than a mimic of body posture or the repeated use of a specific word. Nevertheless, accommodation effects have been found in speech-related features, like speech rate (Local, 2007; Levitan and Hirschberg, 2011) and pitch contour (Babel and Bulatov, 2012).

The automatic utilization of accommodation strategies by speakers makes it an integral part of human communication. However, in recent years, the everyday usage of voice-activated devices has been consistently increasing. This kind of interaction introduces new challenges and coerces humans to adjust their verbal communication to cope with the limited capabilities of computer-based interlocutors. While similar accommodation effects have also been found in human-computer interaction in different experimental settings (e.g., Bell et al., 2003; Parent and Eskenazi, 2010; Levitan, 2013), the effects were present mostly on the human side, whereas the systems' accommodation capabilities were considerably more limited at best. Such one-sided adaptation is incongruous with the mutual, dynamic exchanges occurring in human-human interaction. Expanding the effect to computer interlocutors to simulate the aforementioned conversational dynamics still poses a challenge. This challenge comprises both technical and modeling facets. The former deals with the ability of a system to detect phonetic changes in the human's speech and to manipulate the corresponding features in its output speech in real-time, while the latter refers to determining the relationship between a

user's realizations and the way they influence the system's productions. Direct control over synthesized speech is challenging due to the limitations of current text-to-speech synthesis methods, which almost exclusively use a trained model that cannot be modified on-the-fly. Even if such capabilities are achieved, accommodation models are required for establishing the way the system responds to users' speech variations. This involves some design decisions. For example, should the system try to simulate behaviors observed in human-human interactions or simply follow the user's lead? Should the system initiate changes or only react to user variation? Could the course of the interaction be influenced by the way the system adapts towards the user? All these decisions are part of defining the dynamics between the human and computer interlocutors and may change depending on the specific application.

Integration of accommodation capabilities can be especially beneficial for spoken dialogue systems, as they are typically the core of verbal human-computer interaction. Like improvements in other aspects of spoken dialogue systems, vocal accommodation capabilities will contribute to their enhancement toward more human-like conversational behavior (Weise, 2017). Considering the assumptions of communication accommodation theory, interacting with a system that simulates behaviors familiar from human-human interaction should ease the process for the user and ultimately make it more efficient and fluent. Furthermore, with the usage growth of voice-activated devices like personal assistants, accommodative speech would offer additional degrees of personalization for users. For instance, the speech of a personal assistant owned by one user might differ from that of another and could change when encountering a new user – therefore reflecting the adjustments performed by humans. Furthermore, studying accommodation in human-computer interaction sheds light on the way humans perceive non-human interlocutors in social contexts and whether they want to communicate with them in a similar manner as with other humans. Beňuš et al. (2018) show that a computer-based interlocutor gained more trust from human companions when it exhibited some level of vocal accommodation.

This work investigates the building blocks on the way to achieving vocal accommodation in human-computer interaction. These include experiments for collecting evidence of accommodative behaviors in human-human interaction and human-computer interaction, approaches for modeling these behaviors in a computer-compatible fashion, methods for integrating accommodation models into real-time text-to-speech synthesis,

and implementation of a spoken dialogue system that support vocal accommodation. Previous work has addressed these concepts, mostly independently of each other. Levitan et al. (2016) introduce an approach for integrating prosodic-acoustic convergence into a conversational avatar, but without considering different types of accommodative behaviors. Similarly, Beňuš (2014) examines social aspects of entrainment in spoken interactions, but does not demonstrate how those can be harnessed to measure them and develop models. Obviously, the scope of each study cannot possibly cover all topics. However, in addition to the depth of each of these concepts, the connections between them for introducing a complete solution should be considered as well. For example, the manner in which the experimental findings are converted into a model defines the flexibility and degree of variation of the system. It is therefore important to jointly address both the theoretical and technical facets of the topic, as they can benefit each other. On the one hand, the technical capability to manipulate speech needs a modeled knowledge about the possible (and plausible) changes that might occur; and on the other hand, accumulating empirical data without showing how it models the phenomenon in question makes it highly challenging to demonstrate the essence of the captured evidence.

Offering such a comprehensive overview of this multidisciplinary theme and presenting the individual topics in a wider context were the primary inspirations for this work. A further motivation was to suggest a more structural approach to accommodation description in computers, namely a hierarchy of accommodation levels. Each level builds on the previous one and progressively increases the complexity and variability of the accommodative behavior, from direct mirroring of users' productions to independent responses To that end, empirical data is required for observing a range of behaviors, and appropriate computational means need to be utilized to prevent too simple or unnecessarily complex behaviors. This distinguishment between different types of behavior has received little to no attention so far and can help to better define the desired behavior of a system, based on user's expectation and the target application. Lastly, an emphasis is put on the temporal aspect of conversation – and by extension, of accommodation effects – throughout the work, which is often neglected in studies, but provides important insights on the interactions' dynamics.

1.2 Outline

This work lies at the intersection of two communication phenomena, viz. *phonetic accommodation* and *human-computer interaction*. Both of these topics play a role when talking with any kind of *spoken dialogue system*. The challenges in combining them stem from the complexity and variability of accommodation processes and the absence of this inherent human capability in computers. The core parts are structured to reflect the journey from theoretical ideas and empirical experiments, through modeling, and ultimately implementation of potential applications.

Part I introduces the main topics related to this intertwinement of research areas. Chapter 2 provides an overview of the theoretical, social, and linguistic aspects of accommodation in general, and in spoken language in particular. This includes types of mutual variation throughout a conversation and measurement of accommodation in human-computer interaction. A survey of the ways humans interact with machines is presented in Chapter 3. The properties and challenges of verbal interaction with computers are discussed as well. This chapter also introduces spoken dialogue systems, along with their typical architecture and examples of common modern applications of them. Finally, a suggested roadmap for integrating accommodation capabilities into spoken dialogue systems is explained, together with terminology for differentiating some levels of accommodation in computers.

The main contributions are subsequently divided into three parts: Experiments, Modeling, and Application. A series of empirical accommodation experiments are described in Part II, each in a different social context and constellation of interlocutors. Chapter 4 shows vocal accommodation effects and their utilization in real-world *human-human interaction*. Examining these effects in such conversations helps to determine the gaps between the analysis of conversations in the wild and lab setting. Due to the length of these conversations, analyses of both dynamic changes over time and more general classification of speaker type are possible. Chapter 5 presents shadowing tasks combining both *human-human interaction* and *human-computer interaction* contexts. These tasks were carried out in closely controlled experimental settings for direct comparison between the two contexts. Further evidence of accommodation in a different context is explored in a study of vocal accommodation in *singing*. Lastly, a multiparty *human-human-computer interaction* study is outlined in Chapter 6. This more evolved

mix of speakers sheds light on accommodation effects influenced by the addressee of the specific utterance or by the presence of another human interlocutor.

Part III comprises two approaches for modeling accommodation to be used in a spoken dialogue system. A computational model composed of empirically-motivated parameters is introduced in Chapter 7. This model aims to provide a descriptive way to depict accommodation and craft desired behaviors. The approach taken in Chapter 8 is statistical in nature. It identifies different speaker tendencies, even if those cannot be explicitly be broken into specific properties. Nevertheless, these can be used for defining a target behavior for a system. The use of these two approaches in conjunction is also discussed.

Part IV contains implementations of components required for responsive spoken dialogue systems. The technical details of a module linking between the speech input and output of a system are described in Chapter 9. This includes an additional module for handling accommodation-related processes, like detecting variable sounds and running the model, and a brief survey of real-time, on-demand manipulation of synthesized speech, which enables the required control over a system's output needed for realizing accommodation effects. Together with the modeling information and the techniques from Part III, these components are utilized in the system introduced in Chapter 10. The extended architecture, usage possibilities, and graphical visualizations of this system are demonstrated via a use-case display.

1.3 Related own publications

Chapter 5 – **Shadowing in Sung Music and Human-Computer Interaction**	Eran Raveh et al. (May 2020). "Prosodic Alignments in Shadowed Singing of Familiar and Novel Music". In: *Speech Prosody*. Tokyo, Japan, pp. 606–610. DOI: 10.21437/SpeechProsody.2020-124. URL: http://dx.doi.org/10.21437/SpeechProsody.2020-124 Eran Raveh et al. (Mar. 2017a). "Investigating Phonetic Convergence in a Shadowing Experiment with Synthetic Stimuli". In: *Electronic Speech Signal Processing (ESSV)*. ed. by Jürgen Trouvain et al. Saarbrücken, Germany, pp. 254–261. URL: http://essv2017.coli.uni-saarland.de/pdfs/Raveh.pdf
Chapter 6 – **Accommodation in Multiparty Interactions with an Agent**	Eran Raveh et al. (Sept. 2019a). "Three's a Crowd? Effects of a Second Human on Vocal Accommodation with a Voice Assistant". In: *Interspeech*. Graz, Austria, pp. 4005–4009. DOI: 10.21437/Interspeech.2019-1825 Eran Raveh et al. (Mar. 2019c). "Comparing Phonetic Changes in Computer-Directed and Human-Directed Speech". In: *Electronic Speech Signal Processing (ESSV)*. Dresden, Germany: TUDpress, pp. 42–49

Chapter 7 – **Computational Model**	Eran Raveh et al. (Aug. 2017b). "A Computational Model for Phonetically Responsive Spoken Dialogue Systems". In: *Interspeech*. Stockholm, Sweden, pp. 884–888. DOI: 10.21437/Interspeech.2017-1042 Eran Raveh et al. (Sept. 2019b). "Analyzing Phonetic Accommodation in Human-Human and Human-Computer Interactions". In: *Phonetik und Phonologie im deutschsprachigen Raum*. Düsseldorf, Germany
Chapter 9 – **Accommodation Module for Spoken Dialogue Systems**	Eran Raveh and Ingmar Steiner (Aug. 2017b). "Phonetic Adaptation Module for Spoken Dialogue Systems". In: *Semantics and Pragmatics of Dialogue (SemDial)*. Saarbrücken, Germany, pp. 170–171. URL: http://semdial.org/anthology/Z17-Raveh_semdial_0027.pdf Eran Raveh and Ingmar Steiner (Sept. 2017a). "Automatic Analysis of Segmental Features in a Real-Time Phonetic Convergence Pipeline". In: *Phonetik und Phonologie im deutschsprachigen Raum*. Berlin, Germany. URL: http://www.coli.uni-saarland.de/~raveh/assets/pdfs/Raveh2017PundP.pdf
Chapter 10 – **Web-Based Responsive Spoken Dialogue System**	Eran Raveh et al. (Sept. 19, 2018). "Studying Mutual Phonetic Influence With a Web-Based Spoken Dialogue System". In: *Speech and Computer (Specom)*. Ed. by Alexey Karpov et al. Vol. 11096. Lecture Notes in Artificial Intelligence. Springer, pp. 552–562. DOI: 10.1007/978-3-319-99579-3_57. URL: https://arxiv.org/abs/1809.04945

Chapter 2

Phonetic Accommodation

In this chapter, the concept of accommodation is introduced. Types of linguistic changes and the related terminology used in this work are explained as well. Finally, a survey of works on phonetic convergence in human-computer interaction is presented and discussed to lay the ground for the contributions of this work.

2.1 Communication accommodation theory

Communication is a fundamental part of life. In addition to offering a means for exchanging information and expressing emotions and desires, it also exhibits salient markers of social membership. Human communication is a complex concept concealing many facets and sub-processes within it. This Complexity stems from two main aspects: First, each individual is unique and communicates differently – in general as well as across specific interactions – based on inherent traits, emotional state, personal preferences, situational circumstances, and more. Moreover, an interlocutor may belong to or represent a certain group (e.g., a social group or an organization), which may also influence the nature of the exchange. Secondly, some forms of communication, in particular face-to-face, harness combinations of modalities. This results in a large amount of information one needs to process in real-time to achieve efficient communication. Furthermore, despite common social conventions, not every person perceives and processes this information the same way, which requires all interlocutors to be attentive as to how they comprehend others and vice versa. Therefore, each exchange is unique and shaped by various personal and environmental factors, which often makes interactions hard to analyze and predict (ass discussed in Section 3.3.1). To cope with such highly variable dynamics, people must have some way to know – whether unconsciously, intuitively, or deliberately – how to adjust their communicative behavior with respect to the other interlocutors in the conversation.

Communication accommodation theory (CAT) is a theoretical framework of communication which aims to explain the personal and social motivations in verbal and non-verbal human communication. A core motivation in CAT is *social distance* (Giles, 1973; Giles et al., 1991; Giles, 2007), which suggests that to reduce social distance, a speaker may converge toward the conversation partner(s), whereas divergence would lead to an increase in social distance. When and how social distance should be altered depends on the social class of the speakers, their formal role and personal goals in the interaction, etc. Particularly, these changes occur with respect to how the other conversation partners speak, from overall psychological, social, and linguistic behaviors to specific features (like those introduced in Section 2.2). For example, different communication styles would probably be utilized when speaking to a childhood friend, a colleague, or a company's executive. In each of these situations, the speakers would likely use different

language registers (e.g., slang words and politeness markers) in an attempt to fit into the social groups and become closer to its members. Another example is the use of "elder language" when talking to old people (e.g., using slower speech, extended use of hand gestures, etc.), to make it easier for them to understand the speaker. Such adjustments can be, to some extent, conscious, but often occur automatically. Ideally, both speakers eventually find their comfort zone during the conversation. However, there exists the notion of over-accommodation, which is usually caused by very conscious speakers that intentionally exploit this phenomenon. If realized by a conversation partner, this might be perceived as pretentious or fake behavior and even mockery. Similarly, accommodation can serve a speaker with audience design when talking to a larger group of people. The adjustments may be reflected simultaneously in different modalities, like hand gestures, body posture, facial expression, eye gaze, lexical choices, and speech. This work concentrates on the latter. While CAT advocates these changes, it is worth mentioning that there are other further frameworks that explain them as well. For instance, the Interactive Alignment Model (Pickering and Garrod, 2004, 2013) offers a model where adjustments in communicative behavior are described as *alignment* as opposed to *accommodation*, hinting that the process is rather one-sided and uni-directional (see Table 2.1). Despite this difference and others, all these frameworks describe a process where the similarity between interlocutors increases or decreases with respect to certain features over the course of their communication. Section 2.1.1 sheds more light on the difference between the numerous terms used to describe this process and how they are used in the literature.

2.1.1 Variation types – terminology

The term "convergence" is a central notion in this work. This term's definition in this work is different from its meaning in other fields, like machine learning (ML). Moreover, various terms are used in the literature synonymously to describe the same phenomenon, although there are often subtle differences or emphases on certain facets of it. Disambiguating these terms should help to avoid confusion and allow a more fine-grained description of such processes. The core meaning of convergence is explained here in detail, followed by a list of related terms and explanations about their use in the literature.

The verb *to converge* is defined in Cambridge Dictionary[1] as "moving toward, or merging into, the same point (e.g., roads)". A more non-materialistic definition is provided as well: "If ideas and opinions converge, they gradually become similar". Another, somewhat more general, definition to the noun *convergence* is given in Merriam-Webster dictionary[2]: "the act of converging and especially moving toward union or uniformity". As these definitions suggest, the core idea of this concept is two (or more) entities that change (potentially in different degrees) toward some physical or abstract point and ultimately meet. In some time-dependent cases, like spoken interactions, there might not be enough time for the speakers to meet, but they can still become more similar to one another nonetheless. This matches the definition of *phonetic convergence* proposed by Pardo (2006): "[…] increase in segmental and suprasegmental similarities between two speakers". This also resembles the definition suggested by Xia et al. (2014): "[…] behaviors become more similar over time", with "behaviors" referring to different modalities of communications in conversation, e.g., facial expressions, gestures, lexical choices, etc. Other terms are sometimes used to describe similar processes or different aspects, and it is important to make the distinction between them. Note that some of the terms are used interchangeably, as synonyms, or as hypernyms and hyponyms in other works.

The list presented here aims to unequivocally distinguish these terms and grant them useful relations to offer a meaningful common terminology, at the very least within the scope of this work and hopefully in the community of this research area in general. Table 2.1 summarizes the terms comparison.

accommodation – is often used in the literature as an overarching term derived from its meaning in CAT and includes any dynamic mutual changes during an interaction. Concretely, it includes both convergence and divergence, but also other forms of change caused by influences of other interlocutors. An important aspect of any accommodative effect is that it is a result of external input.

convergence – As explained above, convergence means an increase in similarity between speakers (see Figure 1 in Levitan and Hirschberg, 2011). Each speaker can account for a different "amount" of the overall convergence effect, based on the

[1] http://dictionary.cambridge.org/dictionary/english/converge

[2] https://www.merriam-webster.com/dictionary/convergence

Table 2.1: Comparison of variation types properties. The *mutuality* column marks whether the process occurs in both interlocutors. *directionality* indicates whether the process is defined in a specific direction (similarity or dissimilarity). *Intention/awareness* imply that the process is performed consciously, and the last column shows whether the process' trajectory advances toward a specific, potentially relative, *target* value or not. A (✓) sign indicates that the property is fulfilled to some extent, or that it may be implied or assumed in some cases. The '*' signs mark the terms most widely used in the literature.

	mutuality	directionality	intention/awareness	defined target
accommodation*	(✓)			
convergence*	(✓)	✓	(✓)	
mimicry	(✓)	✓	(✓)	(✓)
proximity	✓			
synchrony	✓	(✓)		
mirroring	✓	✓		✓
coordination	✓	(✓)	✓	(✓)
assimilation		✓	(✓)	(✓)
adaptation*		(✓)	✓	✓
chameleon eff.		✓		(✓)
priming		✓		✓
entrainment*		✓	(✓)	(✓)
alignment		✓	(✓)	✓
imitation		✓	✓	✓

speaker's tendency to converge in the specific interaction and in general. There may be great individual differences in this tendency, as shown in the experiments in Part II. This tendency is referred to as the speaker's *sensitivity* to external changes in Section 7.3. Convergence can result in the speakers' production values meet somewhere in the middle (as in Figure 2.2) or closer to one of them, depending on the social dynamics in the interaction. Similarly, **divergence** is the reversed effect, i.e., when a decrease in the similarity between the speakers occurs. Divergence is generally less common in human-human interaction (HHI), but may occur in competitive (as opposed to collaborative) scenarios, or when a speaker, intentionally or not, thrives to deliberately be distinguished from the other.

entrainment – is presented by Brennan (1996) as the possibly aware influence of an interlocutor on another, e.g., imposing lexical units in an interaction, with emphasis on the one-sided nature of the process. This means that one of the interlocutors

established the use of an aforementioned term which created a bias by the conversation partner to employ it as well. The change is sometimes seen as categorical (similar to *priming*), which is a key difference from the definition of convergence here. This is especially relevant in human-computer interaction (HCI), as it is much easier for a computer, due to the vocabulary problem (Furnas et al., 1987), to establish such uses. Lopes et al. (2013) describe entrainment as imposed by a human speaker, where a computer-based interlocutor follows the lexical choices of the user. This term is sometimes also used as a static measure of changes in an interaction (cf. Section 2.2.2), as in Levitan (2013). Conversely, *convergence* refers here to the potentially mutual dynamic measure, with the difference being comparing discrete timestamps in the interaction (for example, between two halves of a session, as done by Xia et al., 2014) versus changes occurring gradually over its entire length. In addition to all the above, entrainment is often synonymous with convergence or accommodation in more technical fields.

synchrony – refers to an ongoing process where the interlocutors are changing their behavior similarly, i.e., synchronously. Importantly, this applies to the relative changes in each speaker, but not to any absolute values. That is, the distance is generally maintained while the individual ranges may differ (see example in Figure 2.1 on page 20 and Figure 1 in Levitan and Hirschberg, 2011). When one of the speakers is leading the synchrony, it becomes *lagged*, as shown in Figure 2.2. Lagged phenomena and the speaker leading them are often determined using some correlation measure, like Peasrson's coefficient in Edlund et al. (2009) and Xia et al. (2014). A deeper analysis of such an accommodation effect is presented in Section 4.4.

adaptation – refers to the process of making intentional changes that suit certain conditions or situations. As such, this term stresses the ambition of these aware modifications in vocal behavior made by an interlocutor to be more similar to a known and defined target exhibited by another speaker. Kang (2010) and Hwang et al. (2015) both examine the phonetic adaptation process that occurs gradually when encountering a new sound environment, to which speakers want or are expected to adapt. In these cases, the changes are also likely to be maintained outside the scope of a specific interaction. Adaptation is also used to describe the technical capability – or more often the lack thereof – in a machine to match its speech pro-

duction to a user interacting with the system. This fits the definition of adaptation being intentional and having a well-defined target to adapt towards. However, this is still a very limited feature due to the current state of text-to-speech (TTS) technologies. The integration of such capabilities into spoken dialogue systems (SDSs) and the involved challenges are discussed in Section 3.3 and Chapter 9.

priming – is similar to entrainment, but usually works in a larger scope, in terms of both time and the degree of change. It is typically used in the context of psycholinguistics. As opposed to entrainment, e.g., on the lexical level, priming can influence not just the use of one specific term, but a whole semantic field. For example, it was shown in experiments by Meyer and Schvaneveldt (1971) and Schvaneveldt and Meyer (1973) that people respond faster to words from a specific semantic field after being exposed to other words from it over a long timespan. In more general terms, priming changes the likelihood of a person to use specific behavior, typically on some semantic or syntactic level. The temporal scope is different as well. Priming can have an effect in a longer-term than a single interaction, ranging from multiple interactions across several hours, days, and up to years. Therefore, this term is often used when talking about language change in children (see, e.g., Huttenlocher et al., 2004; Wansink et al., 2012). Like entrainment, priming also usually describes a categorical change (cf. Reitter et al., 2006; Pace-Sigge, 2013).

assimilation – implies a one-sided process, where one side changes in a certain way to match the other. It is typically used to describe an accommodative process with the motivation to associate oneself with a social group by adopting its vocal characteristics. Therefore, assimilation is seen as a change that occurs as a result of a specific situation or context, e.g., public speeches, as shown by Ohala (1990) and discussed in Section 2.2.1. This resembles the use of this term in phonology to describe a sound that changes to become similar to another sound with respect to a certain property (see examples in Hall, 2011, pp. 89-98).

alignment – is a term derived from the Interactive Alignment Model (Pickering and Garrod, 2004) and describes an increase in similarity between speakers. It describes a process similar to assimilation, but with a certain motivation behind it. While in assimilation the change is measured with respect to a social group or speech style, alignment is often used to describe the adoption of speech characteristics of

a speaker in a specific conversation. Specifically, it is claimed to contribute to the overall ease and success of conversations by the interlocutors to behave similarly on different linguistic levels (Garrod and Pickering, 2009).

mirroring – is a cognitive tool that aims to produce an output that follows some input. A.k.a. "mirrored equivalence" (Messum, 2007; Messum and Howard, 2015), this process differs from imitation by the lack of aim toward a well-defined target, but rather an internal representation of the similarity between the speakers. This may lead to a less planned and sometimes automatic effect of becoming closer to an interlocutor, where the specifications of the change do not necessarily match the input target. Less commonly, this term also refers to an effect opposite to synchrony, where the directions of the respective changes are reversed instead of similar (as if a mirror was put between them). Additionally, mirroring may also refer to an effect similar to mimicry, but with greater emphasis on speech learning and acquisition (e.g., Yoshikawa et al., 2003).

mimicry – is the tendency to behave, or speak, broadly like someone else. The emphasis here is on the *general* inclination, which hints it can be done unintentionally, or, alternatively, be deliberate, but without the goal to perfectly match the target (as opposed to imitation). Gueguen et al. (2009) demonstrate how mimicry can earn the mimicker more favorable judgment in social interactions. This is explained by mimicking creating a greater feeling of affiliation and rapport in communication, or with the more positive evaluation of the mimicked person due to an enhanced familiarity established by the mimicker. Parrill and Kimbara (2006) show just how natural mimicry in HHI is using an experiment in which participants' behavior was affected merely by watching mimicking takes place in another conversation.

imitation – is similar to mimicry, but is done intentionally and with the goal to match as closely as possible to a target. In speech, it emphasizes the speaker's intent to completely match the auditory input (cf. Gueguen et al., 2009). The attempt to *deliberately* repeat and *accurately* replicate another speaker's productions distinguishes imitation from mirroring and mimicry.

chameleon effect – is a more general account of mimicry, but with complete non-conscious adoption of an interlocutor's behavior (Chartrand and Bargh, 1999). It is a term from social psychology typically associated with a comprehensive change

in multiple modalities and in the overall mannerisms. Gueguen et al. (2009) focus on the social aspects of this effect and how it can change social judgment and attitude toward the speaker. Both works describe it as ”monkey see, monkey do”, which emphasizes the automaticity of it. Moreover, they state that it is a learning mechanism for children and for humans in general before the development of unified languages (Gueguen et al., 2009, p. 256).

proximity – describes general closeness between interlocutors (as illustrated in Figure 1 in Levitan and Hirschberg, 2011). This does not imply specific distance thresholds or absolute value ranges. Furthermore, this is a rather passive, potentially merely circumstantial, state, which does not involve a defined vocal target or an aspiration to match it. The proximity at a specific point in time (e.g., around the start of a conversation) can be used as a reference point for other measures.

coordination – implies cooperation – either seeming or real – between interlocutors. Increase or decrease of communication features is in this case a side of effect of this coordination. This provides another point of view on the process, namely not to examine the speakers' collaboration based on common changes, but looking at these changes as part of this collaboration.

2.2 Linguistic accommodation

Accommodation occurs on various linguistic levels in HHI. Relative salient changes may occur on the lexical level when one interlocutor shifts his lexical choices to match those of another. This is more likely to happen when a lexical entity has multiple commonly used alternatives, like synonyms or different names. For example, Jucks et al. (2008) show how healthcare experts try to match their wording to patients in written inquiries. In another experiment, Friedberg et al. (2012) found increasing lexical similarity over the course of spoken discussions among students groups with better performances. Rácz et al. (2020) even extended the analysis to morphological forms, suggesting that morphological convergence occurs and creates generalizations in memory in real-time. Other examples of lexical convergence have also been found in HCI when looking for information (Lopes, 2013) or when playing (Bergqvist et al., 2020, and see Section 2.3.2). In all these experiments, it was shown that lexical convergence had a positive effect on task performance.

Figure 2.1: Example of pitch synchrony between two speakers in conversation number 2005 of the SwitchBoard corpus (Godfrey et al., 1992). The turns of speaker A (red; top) and speaker B (blue; bottom) are marked by the respective vertical bars. The circles in the corresponding colors show the individual production values. The locally estimated scatterplot smoothing (LOESS) trend lines show the overall synchrony (with slight convergence) between the speakers over the course of the conversation. The speech balance measure at the bottom is a value between 0 and 1 indicating how balanced is the overall speaking time between the two speakers (the higher the more balanced, and the interactivity measure indicates the frequency of floor change without (and with) backchanneling (0 points to only long monologues and 1 to floor change after every turn). These measures are further explained in Section 4.4.2 on page 71.

The focus in this work is on accommodation occurring in speech-related features, i.e., *vocal accommodation*. A core difference between lexical and vocal accommodation is the mechanism for defining two instances as the same. For example, the word "window" is written the same way regardless of who writes it, which makes it easy to associate tokens with the same type. However, vocal features are strongly dependent on the speaker. The signal representing a word would be different depending on the speaker's physiological properties (like vocal tract size), speaking rate, voice intensity, intonation, and many more. Moreover, it is very unlikely that even the same speaker would pronounce the same word in the same manner. This is especially true in conversation taking place in different settings and environments, but also for two successive utterances in the same conversation. Additionally, the relative differences, e.g., in vowel quality or speaking rate, also differ from one person to another, which entails that more evidence might be needed for a perceivable target for accommodation to emerge. An exception to these differences could be categorical phonetic differences, where each category may be perceived as a separate entity (similarly to the lexical case), making it easier to define the target (and see Sections 5.3.1.1 and 9.2.2). This great variety in spoken language makes the accommodation process more complex, as the targets might not always be well-defined and static.

Vocal accommodation has been found in both segmental (Smith, 2007; Pardo et al., 2010) and suprasegmental (Shockley et al., 2004; Walker and Campbell-Kibler, 2015) phonetic features and both in conversational (Pardo, 2006; Lewandowski, 2012; Weise and Levitan, 2018) and non-conversational (Shockley et al., 2004; Babel et al., 2014) scenarios. There is evidence for it being both an internal mechanism (Pickering and Garrod, 2004) and socially motivated (Giles et al., 1991; Babel, 2010; Kim et al., 2011). For instance, phonetic convergence (Giles, 1973) or divergence (Bourhis and Giles, 1977) is triggered by decreasing or increasing social distance between interlocutors, respectively. Baumann (2020) even shows that feedback given by conversation partners can influence the convergence process. Aubanel and Nguyen (2020) found that speakers accurately and consistently converge to each other's fundamental frequency (f_0) in scripted read-aloud dyadic conversation on a turn-by-turn basis. This shows the ability to track changing f_0 both in perception and production. Similar effects were demonstrated by Pardo (2013) with respect to multiple phonetic features. The study conducted by Babel and Bulatov (2012) supports the importance of f_0 in convergence effects by showing that filtering

out the f_0 frequencies in the signals the participants hear leads to reduced convergence. Schweitzer et al. (2017b) show that convergence effects are stronger when conversation partners can speak but not see each other, and divergence occurs more when they can also see each other while talking. Moreover, the effects were stronger depending on the degree of likability between them. This shows that accommodation can be influenced by other, non-linguistic factors of the conversation. According to Lehnert-LeHouillier et al. (2020), language skill level and greater f_0 expressiveness also influence the degree of convergence. Accommodation effects were also found in intensity levels of speakers: In an experiment where participants heard an interviewer in different levels of vocal intensity, Natale (1975) shows that their intensity was generally changed accordingly. In a second experiment, the degree of intensity convergence could be predicted by a social desirability test, which stands in line with the finding of f_0 accommodation. A third commonly studied feature is articulation rate (AR; or the related measure *speaking rate*). In an exemplar-theoretic view in mind, Schweitzer and Walsh (2016) investigated how syllable frequency influences the degree of change. Evidently, stronger effects were found in more frequent syllables, which supports this view. Relatedly, Edlund et al. (2009), Xiao et al. (2015), and Cohen Priva et al. (2017) introduce ways to measure temporal prosodic changes, like pauses, in conversation, which affect the speaking rate. Local (2007) and Levitan and Hirschberg (2011) found convergence effects in all these three features (and others) in collaborative games and everyday scenarios. Such a wide variety of evidence suggests that, even in the vocal modality alone, accommodation is reflected in various ways in HHIs. These features are investigated in this work in both HHI (Chapter 4) and HCI (Chapter 6) contexts. In addition to these, other factors and phonetic features were investigated for accommodation, such as voice quality (Borrie and Delfino, 2017), voice onset time (Nielsen, 2011), and other timing-related phenomena (Putman and Street, 1984), second language proficiency (Law et al., 2020), interlocutors' sex (Levitan et al., 2012; Bailly and Martin, 2014), perceived attractiveness (Michalsky and Schoormann, 2017), word frequency (Nenkova et al., 2008), and more. A survey of methods for measuring accommodation in HHI can be found in De Looze et al. (2014) and Lewandowski and Jilka (2019).

2.2.1 Long-term and short-term sound changes

All the sound changes discussed in this work are short-term changes occurring over the timespan of a single, even if long, interaction (Chapter 4) or multiple sequential interactions (Chapter 6). Similarly, the models and applications in Chapters 7 to 10 are also designed to handle accommodation-related changes within the scope of individual interactions. However, sound changes may occur continuously over long periods of time – even years or decades. While short-term accommodative changes can be ascribed to local social influences of specific interlocutors (as discussed in Section 2.1), long-term changes may stem from larger cultural influences and independent evolution of a person's speech. The changes' source can be both the speaker and the listener (Ohala, 1989, pp. 176-187) and is typically caused by confusion or correction (Ohala, 1993). The latter is more relevant, for instance, for short-term assimilation, as demonstrated in Ohala (1990). The former, however, is more dominant in long-term changes and cross-language influences, e.g., when sounds in one language are replaced by similar sounds in another in loanwords. This is also related to mutual influences of speakers in the evolution of a language, or the way different speaking styles can be created within a language to mark cultural, regional, and social differences. These reasons and others are explained from the phonetic point of view in Sweet (1874).

Such long-term changes can also occur in the speech of a single person. Harrington (2007) examined vowel changes in the Queen's pronunciation in her annual Christmas messages from 1952 to 2002. Some gradual changes were, indeed, found, but could not be ascribed to age or varying speech style. The author, therefore, considered it an independent, long-term sound change of an individual. Contrarily, in the pronunciations around the 1980s assimilation was found towards accents associated with younger people of lower classes (Harrington et al., 2000a,b). This suggests a potentially aware audience design from the Queen's side (Bell, 1984). Such use of communication falls under the social motivation of CAT, although the interactions were one-sided. In this example, the changes were not a result of interactions over a long period of time. However, this may also happen between people who regularly speak with each other for many years. This is relevant for HCIs that are designed to last a very long time, like personal assistants (PAs) or social companion (see Section 3.2). Therefore, in such systems, accommodation effects should be taken into account as well, but the modeling approach could take advantage of the fact that there is a much longer timespan for the changes to shape. This can be used,

for example, for accumulating more evidence before deciding on the appropriate change from the system side (cf. exemplar approach in Chapter 7 and floor-change approach in Chapter 8).

2.2.2 Measuring accommodation

Conversations are complex processes that require some expertise and quantitative analysis tools for detecting and isolating specific patterns in them, especially since effects may have long, non-linear relations. A behavior is a complex collection of conducts of a person, particularly those towards a certain environment. The realization of one's behavior over the course of a conversation can communicate information regarding one's state and goals, especially with this deviates from the individual's expected behavior. The behavior leads to reactions to environmental input and can be modified due to reinforcements from the environment or self-directed motives. These modifications can occur over time quickly or slowly, consciously or unconsciously, and to a greater or lesser extends, which makes them dynamic and thus cannot be defined discretely. All the above is true for vocal behaviors as well, which are expressed in spoken interactions. Specifically, vocal accommodation reflects dynamic changes during a conversation that can be affected by the external speech input of other interlocutors. It can therefore be beneficial to **move away from value comparisons in favor of behavior descriptors** to depict an interaction. This is done by analyzing spoken interactions as entire events with a continuous temporal dimension as opposed to a comparison between discrete points in time. Examining the whole conversation can help, for example, to determine who was leading the changes or when more accommodation occurred. In this work, such analyses were utilized in Chapter 4 to determine the leading speaker in each conversation and in Chapter 6 to expose the ongoing changes in the human speaker's productions.

The way speakers accommodate to each other is very unlikely to be linear from beginning to end and can change throughout the conversation. Therefore, comparing the differences between two discrete, distant points in time might miss or oversimplify some dynamics that occurred between them. Moreover, comparisons like that usually take the view of one speaker instead of looking at the conversation as a complete entity. Quantitative analyses often leave accommodation hidden or overly smoothed if done on a turn-by-turn basis or by splitting it arbitrarily into two or more parts (often equally-long, non-overlapping time intervals) and directly comparing them using raw values as in

Heldner et al. (2010), Rahimi and Litman (2018), and Ibrahim et al. (2019). De Looze et al. (2014, p. 15) point out that in these cases it is assumed that accommodation is a strictly local phenomenon where a speaker's utterance is linked exclusively to the other interlocutor's immediate preceding utterance. Such analyses result in a linear, static representation of the conversation's evolution, from which generalized conclusions are hard to draw. They might even be inaccurate or misleading, especially when both speakers change their output over time, as demonstrated by Cohen-Priva and Sanker (2019). That work refers specifically to difference-in-difference (DiD) measurements, where pairs of two discrete points in time are compared to measure accommodation between speakers using the following distance formula[3]

$$DiD_{\vec{i},\vec{s}} := \sqrt{(\vec{i}^{n}_{t+1} - \vec{s}^{n}_{t+1})^2 + (\vec{i}^{n}_{t} - \vec{s}^{n}_{t})^2} \equiv \|\vec{i} - \vec{s}\|_n. \tag{2.2.1}$$

Figure 2.2 shows the interaction between two speakers' productions in a hypothetical conversation. By merely looking at the plot, it is clear that the two speakers do not sustain the same behavior throughout the conversation. For example, between marks A and B the orange speaker's values go remarkably downwards (though not linearly), while the green speaker remains roughly stable – i.e., *divergence* (although this can also be seen as an independent change). Subsequently, the distance is generally maintained between B and C. Between C and D *convergence* occurs **in both speakers**. Lastly, lagged *synchrony* can be observed between marks D and E. If compared directly, the lagging might make the value changes look somewhat random. However, looking at the entire segment, it's clear that the change is similar and is led by the green speaker. None of these mutual behaviors can be captured by a DiD-based approach. For instance, comparing the beginning (mark A) and end (mark E) of the conversation would lead to the conclusion that there was no change in the values (illustrated by the corresponding dashed lines). Similarly, splitting the conversation into two halves (A to D and D to E) would make it look like the changes were symmetrical, missing the obviously different behaviors of the speakers in these two halves. Additionally, the directionality of the

[3]Note that this way of measuring DiD is commutative and therefore doesn't measure the changes from the view of a specific speaker (unlike, e.g., Cohen-Priva and Sanker, 2019, p. 3).

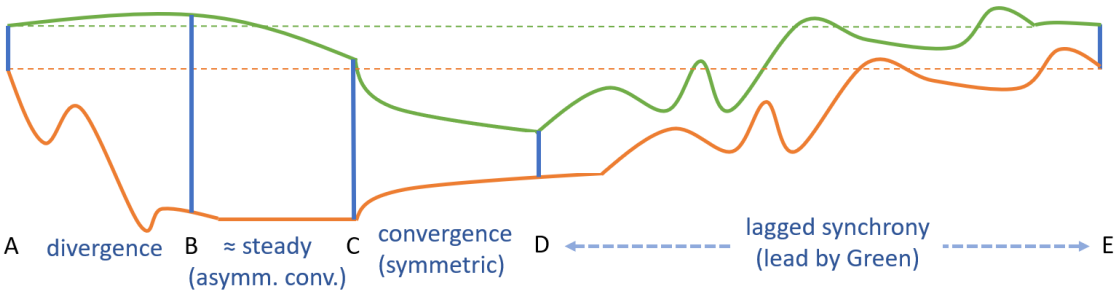


Figure 2.2: The evolution of a feature's values produced by two speakers represented by the green and orange solid lines. The dashed lines connect the corresponding initial and final values of each speaker. The letters A to E mark timestamps with behavioral changes. Each caption describes the behavior between the two corresponding marks.

convergence between marks C and D will be missed by simple DiD distance measures, which might give the impression that the changed rooted from only one of the speakers. This could be satisfactory when it can be assumed that such changes can only occur in one speaker, like when talking with some computer-based interlocutor like a PA (see Sections 5.3 and 6.2), but not when both speakers' productions may be flexible, like in HHI (Section 4.4) or ultimately adaptive spoken dialogue system (Section 3.3). This limitation can be compensated to some extent by examining the *relative* occurring changes *gradually* (Figure 6.8 and Section 6.4.2).

2.3 Vocal accommodation in human-computer interaction

2.3.1 Verbal interaction with computers

A human-human interaction (HHI) is a mutual or reciprocal relationship between two (or more) interlocutors within a limited timespan. This is also true for interaction with machines, though the beneficial side is typically the human speaker(s) while the machine is used as a tool to achieve the humans' goal. In more modern applications, and especially when artificial intelligence (AI) is involved, the computer might also be programmed to "benefit" from the interaction as well, e.g., by acquiring information for future interactions or being able to finish a task more efficiently. One type of interaction is a *conversation*, where the communication is language-based. This difference is more prominent in HCI, since there are ways to interact with machines without using written or spoken words, like using touchscreens, a computer mouse, or hand gestures. This can

be compared to non-verbal – neither written nor spoken – human communication, but purely non-verbal communication occurs more often with machines than with people. The main reason for that is probably since machines are not, yet, capable of using language as freely and verbosely as humans. The terms "interaction" and "conversation" are often used interchangeably in this work, since the interactions in question are spoken conversations. Nevertheless, interactions refer to a more general concept of communication that might involve other components than speech while conversations focus on the language components of the communication.

Interestingly, humans almost always need to compromise on the way they interact with computers or learn new interfaces (like those mentioned above) even for performing simple tasks. Even in the case of spoken communication, which develops early on in humans, compromises need to be made as to how to speak to the computer so that it understands the user's intention. For any of the system types mentioned in Section 3.2, users need to learn how to modify their speech so that they can properly use the systems (be it the speech style or wording), instead of the system being able to adapt to the user. With the advances in speech technologies, this gap is shrinking, but there is still a way to go before computers will be able to understand and produce spoken language well enough for people to speak to computers the same as they speak to other human beings. The topics in this work capitalize on this evolution, to see whether HHI phenomena like vocal accommodation are transferred to HCI as talking to computers becomes easier and more common. More generally, the question arises whether CAT holds for HCI as well. This is supported, for example, by the Computers Are Social Actors (CASA) paradigm (Nass et al., 1994; Nass and Moon, 2000), which argues that humans apply similar social behaviors when interacting with computers because they ascribe human characteristics to them. As a starting point, domain-specific systems with alternate turn-taking are easier for computers, as they take away a lot of the complexity of spoken language and reduce it to individual utterances that can be mapped to actions the systems support. For example, a SDS for ordering train tickets will probably follow a very specific protocol and react only to a specific input, as opposed, for example, for a general-purpose system that could talk with the user about a planned trip and help booking tickets as part of a longer, general-purpose conversation (see Table 3.1).

2.3.2 Previous work

Section 2.2 discusses vocal accommodation in HHI, but this phenomenon has been studied in HCI as well. The key difference between the two settings is the lack of inert changes in computers. Since accommodation is often ascribed to mutual social aspects (as explained by CAT in Section 2.1), this introduces a limitation on the computer's side. Two main approaches are used to overcome this limitation in experiments: Simulating a computer's output in a wizard-of-Oz setting and integrating basic accommodation capabilities into a SDS. Wizard-of-Oz experiments have the advantage that the output of the computer-based interlocutor can be directly controlled by the experimenter, usually using pre-defined utterances. This grants precision and control over the experiment, which makes it a very suitable approach for research. However, preparing the experimenter's control interface and the utterances might be time-consuming, e.g., if they need to be recorded or manually manipulated in a certain way. Another disadvantage is the disability to deviate from a pre-defined script covered by the prepared utterances, limiting the variety of interactions the simulated system can support. SDSs that support at least some level of accommodation or real-time manipulation save the time and effort of creating stimuli prior to the experiment. Though the quality of the output and the time required to generate it might be affected, this setting better represents real-world HCIs and can be more flexible in different scenarios. Just like real-world systems, it requires a lot of time to develop SDSs with these capabilities, which often makes this option impractical for research. Section 3.3 discusses further facets and possible solutions for integrating accommodation capabilities into SDSs.

Using these two methods, various experiments have been conducted to measure accommodation in HCI. Lopes et al. (2011) and Bergqvist et al. (2020), for example, focused on dynamic entrainment and adaptation on the lexical level and found that users adapt to a system's terminology that differs from theirs. This also led to improved performance in the given tasks. Parent and Eskenazi (2010) examined the correlation between lexical choices and word frequency using the *Let's Go* SDS (Raux et al., 2005) and found that users adapt more to words that occur more often. While these studies addressed the changes in experimental, scripted scenarios, the theoretical foundations for studying these changes in spontaneous dialogue exist as well (Brennan, 1996). Levin et al. (2000) and Gašić et al. (2013) provide examples of online adaptation for dialogue policies and strategies. Noticeably, while all the studies mentioned above examined var-

ious facets of dialogues, none of those are related to the auditory aspects of speech – the primary modality used to interact with SDSs – but other did: Beňuš et al. (2018) found relationships between the level of users' trust toward an avatar and the degree of the system's vocal entrainment or disentrainment. Similarly, Levitan (2014, pp. 142-144) shows relationships between prosodic entrainment and how much participants liked the avatar they were interacting with. Bell et al. (2003) found that users' speech rate can be manipulated using a human-simulated SDS. Similar results were found when intensity changes in children's interaction with synthesized output were examined (Coulston et al., 2002). All these experiments focus on HCI, while those in Section 2.2 concentrate on HHI. However, accommodation in HHI and HCI has not been directly compared within the same interaction, as done in Chapter 6. Furthermore, mainly suprasegmental characteristics have been studied for accommodation in HCI, mostly due to technical limitations (see Section 3.3 for details). A wizard-of-Oz experiment with a focus on *segmental* features is described in Chapter 5.

Chapter 3

Spoken Dialogue Systems

This chapter gives an overview on spoken dialogue systems, including common architectures, different system types, and implementation techniques. The concept of adaptive spoken dialogue systems, which is a core topic in this work, is introduced as well, along with the challenges involved and examples of possible adaptation strategies on different levels.

Spoken dialogue systems (SDSs) offer a wide range of services and are used on daily basis in various forms, both for commercial and personal purposes. The main difference between them and other ways to communicate with computers is the use of speech – and mostly speech alone – for interaction. This offers benefits to the users, like being able to perform tasks while keeping their hands free, contrary to systems that require textual input from a keyboard or haptic touch on a screen. We are witnessing an ever-growing presence of voice-activated devices, like speech-activated cars, hands-free medical assistants, and intelligent tutoring systems (ITSs). These devices support more and more functionalities in a way that is more comfortable and intuitive for users. It can be expected that in the near future such devices will be used not only by individuals but also in more social contexts, including interactions where multiple humans are involved. This makes the understanding and improvement of social skills in SDSs all the more important.

The common architecture of SDSs is explained in Section 3.1, along with details about each component and how it can be implemented. Section 3.2 gives an overview of applications that use a SDS at their core to communicate with users. A roadmap toward SDSs with vocal accommodation capabilities as well as the challenges involved in that are discussed in Section 3.3. Ultimately, such capabilities would improve the personalization and overall experience of the interaction.

3.1 Architecture of spoken dialogue systems

As shown in Figure 3.1, the architecture of a SDS is symmetric in terms of input and output types. Each cycle starts and ends with speech signals, generated first by the user, and then by the system (more sophisticated systems can also take the initiative). The content of the utterance, usually referred to as *intent*, is then extracted to determine the utterance's objective. Similarly, the system's speech output is based on generated content that captures some intent. The "brain" at the core of the cycle decides what intent is most suitable for the user's input. This can be done purely by learning from provided dialogue examples, using the help of external information or databases, based on hand-crafted rules, or some combination of those. This simplified flow assumes that the user and the system take turns alternately, one at a time. However, one interlocutor may of course need multiple consecutive turns to convey the message due to length or no response from the other interlocutor. Although each component is a whole research area

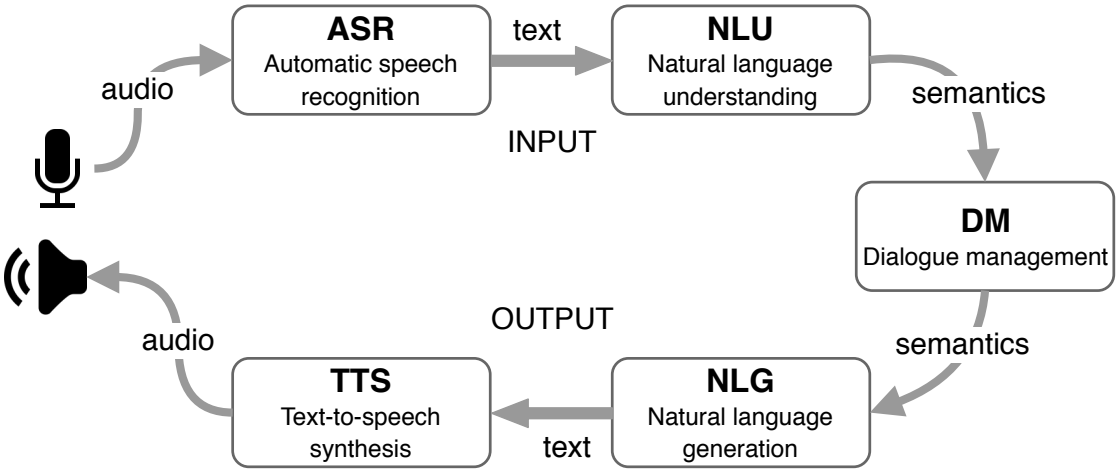


Figure 3.1: A typical architecture of a spoken dialogue system. The interaction lifecycle is symmetric, and for each analysis input step there is a corresponding generation output step. The exchange usually starts with a user spoken utterance and ends with the system's spoken response.

by itself, there are numerous open-source implementations that help to quickly build a basic fully functioning system and focus on a specific one. Brief overviews of these natural language processing (NLP) tasks are given in Sections 3.1.1 to 3.1.5. Note that this is a basic typical architecture and each component may be extended or modified. Specifically, this architecture changes substantially in fully neural-based systems. However, even in that case, the general flow (and by extension the training data) remains the same.

3.1.1 Automatic speech recognition

An essential condition for verbal communication is to be able to hear what the interlocutor says and process it into words. For computer, this is done using automatic speech recognition (ASR), which translates the audio signals produced by the user's articulators into a machine-readable form that can subsequently be fed to the natural language understanding (NLU) component. This step is crucial for vocal accommodation, as it is the only component that accesses the audio signal. However, SDSs predominantly merely use it to extract the said words and discard it afterwards. As a result, they know *what* was said by the user but not *how* it was said. For any kind of responsive behavior, this component must be extended to provide some additional information about the input speech.

3.1.1.1 Tools

Kaldi (Povey et al., 2011) and CMU Sphinx (Lamere et al., 2003) are free-to-use ASR engines that offer various functionalities, including training a model on a custom dataset. For the purpose of vocal accommodation, one benefit of using such modifiable toolkits is the ability to access the phoneme times. This is crucial for detecting and tracking certain phonetic features, and segmental ones in particular (as in Chapter 9).

3.1.2 Natural language understanding

After getting the words uttered by the user, the system needs to infer an intention from them, i.e., what the user wanted to achieve in that turn. This is the role of the natural language understanding (NLU) component. An intention can be as simple as asking the SDS to perform a task (e.g., to turn on the radio in a voice-activated car). such requests are mostly recognizable by pre-defined keywords the system can look for in the transcribed input text. Other requests, like inquiring information about a place or booking a flight, require the system to be able to withdraw – and properly formulate – information from some source. Such tasks require further information (e.g., flight origin and destination, date, price range, etc.) to be obtained, or, if that information is not provided by the user, additional turns where the system asks for the missing bits. This process is called slot-filling. More complex reactions, and especially in the case of chatbots, demands deeper semantic analysis, as the intention might not be explicitly articulated and in some cases, such a defined intention may not even exist.

3.1.3 Dialogue manager

After completing processing the user's input, a decision must be made by the SDS as to how to react. This is done by the dialogue manager (DM), which is the central component of a SDS. The DM is typically divided into *belief tracker* and *policy* modules. The former accumulates information regarding the user's wish based on current and previous turns, while the latter is responsible for determining the most appropriate response to that intention. If any additional data is required for satisfying the user's needs, e.g., some information from the web or a database, the retrieval will be done by the DM. The same goes for specific domain knowledge, which can be made available to the system. Deciding on the best action can be achieved using a deterministic rule system for a small

number of simple cases (a command and control (C&C) system, for example), but need more sophisticated models for more involved situations. One commonly used technique is reinforcement learning, which is suitable for making selecting an action based on a given state. All these conditions determine how flexible and elaborate the system is, and specifically what domain(s) it can handle.

3.1.4 Natural language generation

After deciding on the most appropriate response to the user, the system needs to convey it in a human-understandable manner, namely words. The process is reversed to NLU, i.e., generating text based on a given intent. Depending on the user's input intent, the system may respond with simple acknowledgment statements, repetition and information approval, or completely newly formed full sentences. Additional challenges of this task often root from things that could be reduced or ignored in NLU, but must be precise in NLG. For example, using wrong verb conjugations and tenses will cause the system's response to sound ungrammatical or ill-formed in some cases, but could even lead to a misunderstanding of the system's response. Therefore, depending on the language, the NLG component should be aware of the user's gender, the number of users speaking to it, the nature of the user's intent and how it may be carried out, and more (in multimodal systems, these may influence other modalities as well).

3.1.5 Text-to-speech synthesis

The last step in the flow is converting the text provided by the NLG component to speech signal and play it to the user. This is performed using a text-to-speech (TTS) module, which takes orthographic forms of words and outputs a voice that utters them. Traditionally, voices are learned from recorded human speech by selecting and concatenating small units of it. Linguistic analyses are performed to translate the orthographic forms to sound sequences, insert stresses and pauses, etc. Additional properties, like the contour and duration, are usually determined in inference time. Newer methods are mostly neural-based and can generate audio frames directly from text (e.g., Shen et al., 2018). All these methods have the limitation of not being able to control the generation process directly in each utterance, especially not on the segmental level. This makes it hard to apply detected changes in the user's speech, which is a major barrier on the way to integrate accommodation capabilities into SDSs. Nevertheless, there are examples of

SDSs that can adapt on various levels, including specific modifications in speech (see Section 3.3).

3.1.5.1 Tools

Free TTS engines for training voices include Festival (Black and Taylor, 1997), espeak (Duddington, 2012), and MaryTTS (Schröder et al., 2011). In addition to being used as complete TTS pipelines, these systems can also provide intermediate analysis outputs like phonetic transcriptions.

3.2 Types of spoken dialogue systems

By and large, SDSs can be divided into two main categories, which determine the communication style and behavior of the system: task-oriented systems (e.g., Wen et al., 2016; Zhao and Eskenazi, 2016) and chatbots systems (e.g., Vinyals and Le, 2015; Li et al., 2016). The former has a well-defined scope and aims to achieve a specific goal, while the latter is open-ended with no specific task in mind other than sustaining the conversation with the user. Table 3.1 compares these two system types. Vocal accommodation is relevant for both these system categories, but in different ways. Task-oriented systems may need to accommodate faster and introduce changes more frequently. It might also be required to reset the system's speech for each interaction if it's used by more than one user or for different purposes. Chatbots, on the other hand, might be able to exploit the fact that they are usually involved in longer conversations, giving them more time to learn the user's vocal behavior. This could lead to a slower, smoother process, which should gradually improve the personalization of the system. Another category is voice-activated command and control (C&C) systems, which are arguably not SDSs per se, since they only rarely engage in conversation or trigger a multi-turn dialogue. Therefore, such systems leave little room for accommodation to occur. Nevertheless, C&C systems are considered a simple kind of task-oriented SDS in this work, as ultimately they are designed to achieve a specific task, even if a dialogue is not necessarily required for that. Indeed, task-oriented systems like personal assistants (PAs) often offer C&C interfaces as well. SDSs can be utilized in various ways and be embedded in different types of systems. Sections 3.2.1 to 3.2.5 survey some of the main system types with a SDS at their core.

Table 3.1: A comparison of some characteristics in task-oriented SDSs and chatbots.

	Task-oriented	**Chatbots**
Goal	Help the user achieve a specific, pre-defined goal	Converse as naturally and continuously as possible
Applications	Personal assistants, C&C systems, in-car voice-activated systems, reservations, etc.	Free-form conversational AI applications: chitchat bots, social robots, etc.
Domain	Domain-specific and/or multi-domain	Domain-free or robust cross-domain
Modeling	Statistical models and/or hand-crafted rules	Typically sequence-to-sequence (seq2seq) models with no-go filters
Evaluation	Task completion rate and completion time, number of turns (+ subjective criteria)	Chat length, relevant replies ratio, user engagement, general user satisfaction

3.2.1 Personal assistants

A personal assistant (PA; also *intelligent personal assistant* or *virtual personal assistant*[4]) is a software-based program embedded into a dedicated device (such as smart speakers, see below) that in some way fills the role of a human-being personal assistant. More often than not, this includes mainly straightforward tasks the human assistant can perform, like managing schedules and tasks, but the support for more complex tasks is rapidly increasing and nowadays may also include in-context question answering, smart online shopping, and more. An advantage of PAs is their simple operation, which is almost exclusively voice-based, making them accessible to the general public. Commercial voiced-based PAs include Amazon Alexa, Apple Siri, Google Assistant, Microsoft Cortana, and many other, less famous ones. In recent years, the market for commercial PAs has grown rapidly. For example, Microsoft Cortana had 133 million active users in 2016 (Osborne, 2016) and Echo Dot was Amazon's best-selling product between 2016 and 2018 (Dickey, 2017). Furthermore, 72 % of people who own a smart speaker say

[4]The term *virtual assistant* is widely used as well. However, it is avoided here, since it also refers to a different kind of occupation (cf. https://www.investopedia.com/terms/v/virtual-assistant.asp).

they often use their devices as part of their daily routine (Kleinberg, 2018).

Besides making the operation of such voice-activated systems simple and user-friendly, PAs also aim to let users interact with them in a comfortable, natural manner. One property of natural interactions is the tendency to accommodate to the specific situation and interlocutors to make the interactions more fluent and efficient (Gallois and Giles, 2015). Linguistic accommodation is one aspect of this phenomenon, and it is found in various human-human interaction (HHI) experiments (e.g., Pardo et al., 2017; Schweitzer et al., 2017a). Chapter 6 presents a study of vocal accommodation in multiparty interactions with a PA.

3.2.2 Smart speakers

Smart speakers (or *intelligent speakers*) are small loudspeakers typically used by one to several users in a common household or working environment. The speakers themselves merely offer audio transmission with some basic, hands-free operation. The "smart" portion comes from the software installed on it, which is usually some variety of a PA (see Section 3.2.1). Mainstream smart speakers device series (and the PA powering them) include Amazon Echo (Alexa), Apple HomePod (Siri), Google Home (Google Assistant), and Microsoft Invoke (Cortana). Newer devices, called *smart displays*, can also be operated via a touchscreen. Their easy operation and the convenience they offer make smart speakers very popular with steadily increasing user base, with some estimations of more than 150 million units sold in the United States alone by the beginning of 2020[5] and a rapidly increasing usage in other countries[6].

3.2.3 Chatbots

Chatbots (a.k.a. *chatterbots* and *chitchat bots*) are conversational agents that do not aim to accomplish a specific in-domain task, but to create a human-like communication with the user in lieu of a real human interlocutor. This makes the scope and evaluation of a chatbot more complex, as the definition of the end-goal is not as well-defined (and cf. Table 3.1). Due to their nature, chatbots can be utilized in a variety of ways, and are

[5]https://marketingland.com/more-than-200-million-smart-speakers-have-been-sold-why-arent-they-a-marketing-channel-276012

[6]https://www.emarketer.com/content/global-smart-speaker-users-2019

usually embedded in social robots, virtual agents, or smart speakers that offer one as a separate functionality.

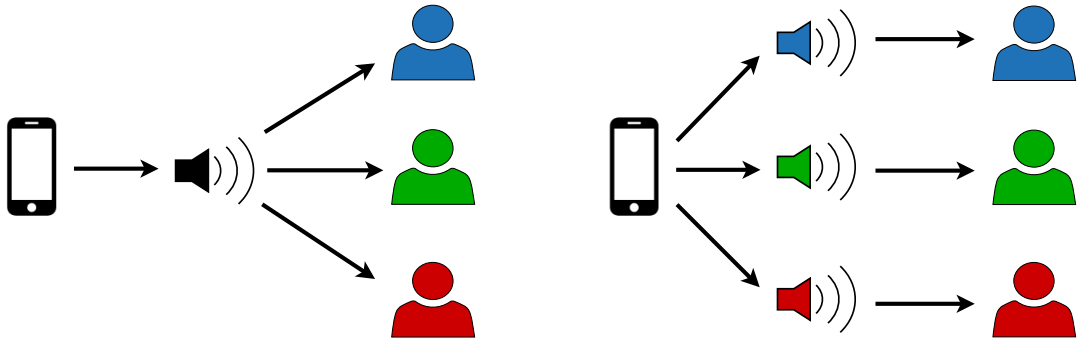
An early example of a program considered a chatbot with a defined purpose is ELIZA (Weizenbaum, 1966), which tried to imitate the role of a therapist in a therapeutic session. While ELIZA's functionality can, for the most part, be reduced to simple word matching, it was revolutionary at the time and opened the way to more sophisticated methods. Nowadays, chatbots are used for improving experience and service in online customer support and instant messaging apps. They have already been used in various domains, such as education (Kerly et al., 2007; Benotti et al., 2014), elderly care (Iio et al., 2020), cultural heritage (Pilato et al., 2005), healthcare (Kowatsch et al., 2017), software development (Lebeuf et al., 2017), and others (Shawar and Atwell, 2007).

3.2.4 Embodied agents and social robots

Embodied agents (sometimes also *interface agents*) are communicative systems with some visual form. Though the embodiment may be graphical only (see Section 3.2.5), this term usually refers to systems that interact and communicate with the environment through some physical shape. For social robots, this shape is normally human-like and may include a full-body representation, like the NAO robot (Singh and Nandi, 2016) or only a face, like the Furhat robot (Al Moubayed et al., 2012). Social robots gather information in different modalities, like eye gaze and hand gestures, and may even generate some limited behavior in these modalities, but ultimately their primary means of communication is almost always speech. Accommodation towards embodied interlocutors is especially interesting, as it is closest to face-to-face HHI. Vocal accommodation has been found in human-robot interaction, e.g., by Ibrahim et al. (2019).

3.2.5 Virtual humans and avatars

Though commonly used interchangeably in the literature, virtual humans (VHs) and avatars refer to two similar yet distinct concepts. On-screen representations of an interlocutor are largely referred to as *avatars*. Those can be static images associated with specific speakers (as in Cohn et al., 2020), but nowadays normally include at least some basic facial expressions and animations. In addition, avatars are also used sometimes as a general term for any virtual, graphically rendered interlocutor (including a VH). VHs, on the other hand, are fully depicted humanoids that aim to portray a real hu-



(a) An illustration of a device's output, which is *not* personalized for each user. The same spoken output (in black) is played to all three users.

(b) An illustration of a device's output, which is *personalized* for each user. A different, customized spoken output (in blue, green, and red) is played to each user.

Figure 3.2: Schematic comparison between an interaction with static spoken outputs (on the left) and an interaction with adaptive (i.e., personalized) spoken outputs (on the right). The system that adapts to the user makes the interaction tailored to the user's behavior.

man being as closely as possible. This typically entails characters with full bodies, but partial figures are used as well, depending on the application. Similar to agents with a physical embodiment, these conversational agents are capable of multimodal communication. They can be used in various interactive activities, like language assessment (Peterson, 2005) and therapy (DeVault et al., 2014). Experiments have shown vocal accommodation effects in interactions with VHs with respect to features like speech rate and fundamental frequency (Staum Casasanto et al., 2010; Gijssels et al., 2016).

3.3 Accommodative spoken dialogue systems

There are various motivations for speakers to change their behavior during spoken interactions. This change can happen in different ways and on multiple levels (Shepard et al., 2001; Gallois and Giles, 2015, and see Chapter 2 for more details), and the changes are mostly driven by external input of other interlocutors. Many experiments have shown that vocal accommodation occurs in HHI (and see Chapter 4), but only in recent years similar experiments were conducted and found similar effects in human-computer interaction (HCI) as well (and see Chapters 5 and 6). However, while in HHI the accommodation is, in its nature, mutual, in HCI only the human speaker could adjust

voice characteristics. This typically results in one-sided, weaker effects, as there is no counterpart to reinforce the process. A major next step would be to add accommodation capabilities to SDSs and make it mutual in HCI, too. According to Weise (2017), integrating these capabilities into computer systems will enhance HCI and provide improved tools for studying accommodation also from the computers' side. He also notes that accommodating dialogue systems could offer an additional layer of interaction, namely responding to user's behavior – either by observing user's behavior and use it as additional information for some analyses or by actively making use of accommodation to encourage the user to speak in a specific way. Though done in a more planned and direct way than in HHI, this offers another level of dynamics from the system's side without changing the interaction's content. Oviatt et al. (2004) discuss the advantages of systems that dynamically adapt their speech output to that of the user, and the challenges involved in developing and using these systems.

Dynamically changing the voice is a capability currently ascribed almost exclusively to humans and exist only sparsely and simplistically in computer-based systems. As illustrated in Figure 3.2, voice-activated devices always talk the same way, regardless of how the user speaks to them, the environment and setting in which the interaction takes place, the goal and role of the devices, etc. This capability, which comes naturally to humans in social interactions, involves several steps of (partially unaware) decisions, which together form the overall effect of becoming behaviorally more or less similar to an interlocutor. These include both situational and knowledge-related facets like how it is expected to behave in certain situations and which vocal changes fit those, as well as the physiological ability to apply these matches. Humans can perform all these steps as one conduct. For computers, however, these steps must be broken down, as they lack the common social background knowledge and the intuition as to how to match their voice to the situation. Ultimately, SDSs with such personalized speech style may offer more natural and efficient interactions, as shown by Porzel et al. (2006), and advance one more step away from the *interface metaphor* (Edlund et al., 2006) toward the *human metaphor* (Carlson et al., 2006), to utilize new user approaches in spoken HCI.

A roadmap towards such systems is discussed in Section 3.3.2 and terminology describing the different levels and ways to achieve accommodative vocal behavior in machines is introduced in Section 3.3.3. Finally, examples of existing accommodative SDSs are presented in Section 3.3.4.

3.3.1 Dialogue is hard

As described in Section 2.2, accommodation occurs naturally in HHI as an integral part of dialogues. The social conventions and unwritten rules of dialogue are implicitly learned by humans, who intuitively know how to apply them and when they can be altered. This is an automatic process that is still too complex for integrating into computers. Therefore, the way we interact with computers is often immensely different from communication with other human beings (Section 2.3.1), making it hard for computers to handle. Another reason dialogue is hard is the infinitely many possible trajectories of each measured aspect. It is always possible to create a dialogue that hasn't been produced before, even concerning a single aspect like chosen words, floor change, vocal properties, length, or any feature of any modality. As if language understanding isn't a hard enough task for computers, the relations between all these aspects and their influence on one another make this task even more complex (Gordon et al., 2018). The complexity of this task is long acknowledged and it was posed by Turing (1950) as an AI-complete problem. Therefore, it is assumed that substantial, maybe human-like, intelligence is required to address its unexpected circumstance, and no specific algorithm or machine learning (ML) method can solve it alone (see further reasoning and examples in Shapiro, 1992, pp. 54-57). Notably, many aspects need to be explicitly split into separate steps and rules for the computer, although human process them as one concept (or at least don't consciously divide it into smaller sub-tasks). One example of this is that humans don't strictly distinguish between task-oriented and chitchat dialogues (Table 3.1), whereas modeling approaches do. Humans can also switch between the two, as they use both as part of one dynamic dialogue. For example, people are good at having an off-topic small-talk at the beginning of a business meeting before switching, gradually or not, to the goal topic (cf. discussion about conversation structure and speaker role in Section 4.1.1). Furthermore, humans intuitively know how to behave based on the situational context, like when to take the floor, when to stop talking, when to barge in, etc., without the need for explicit signals from the other interlocutor. Although there are approaches for teaching computers these dynamic changes (e.g., Skantze and Schlangen, 2009), the gap is still substantial.

As a process that happens as part of a dialogue, accommodation involves some of these challenges (e.g., there may always occur pattern never seen before), but within a limited scope. Moreover, accommodation entails the additional challenge of no "cor-

rect answer", i.e., there is no measure of good or bad accommodation in a dialogue. Humans might be able to point out if an interlocutor uses accommodation in an undoubtedly wrong way (for example, continuously perfectly imitating or demonstratively attempting to talk differently), but probably wouldn't notice how and whether their conversation partners accommodate to them. This poses difficulties in both evaluation and learning, because it is not possible to put labels on accommodation measures without making numerous assumptions. Therefore, the modeling approaches present in this work (Chapters 7 and 8) concentrate on the *behavior* of a speaker within a conversation rather than modeling how well accommodation was utilized. The approach suggested in Section 3.3.2 can be taken as a general scheme for any kind of mutual dynamics in HCI, which is here applied to vocal properties.

3.3.2 Suggested roadmap

The lack of accommodative speech in computer-based systems roots from what is more often than not natural and even automatic for humans, namely realizing how and how much to change their vocal behavior, the physiological means to express those differences, and the ability to combine the two into a coherent production in an interaction. Figure 3.3 shows an overview of a schematic roadmap to integrating adaptive capabilities into a SDS. In addition to the standard functionality of the SDS, three main elements are required: First, knowledge about the nature and properties of accommodative behaviors in humans is required. This includes both empiric experimental data and integrable models. Furthermore, the technical capability to control the speech output on demand is essential for introducing flexibility in the system's base voice. This also includes a mechanism for accumulating phonetic evidence from the user's input relevant for the feature representations used for the accommodation process. As these manipulations must be applied in real-time, re-training the TTS model to capture every change is not only insufficient but not practical as well. This means that the manipulations are done on top of the existing TTS model, either by modifying the outputted waveform directly or by training a model that can consider specific changes in feature description (as done in Section 9.2.2). To link between the modeled knowledge and the audio processing implementations, an additional component must be introduced in the system. The role of this component is to feed the system's flexible voice parameters result from the models to express vocal changes, which are ultimately conveyed to the user. This emphasizes

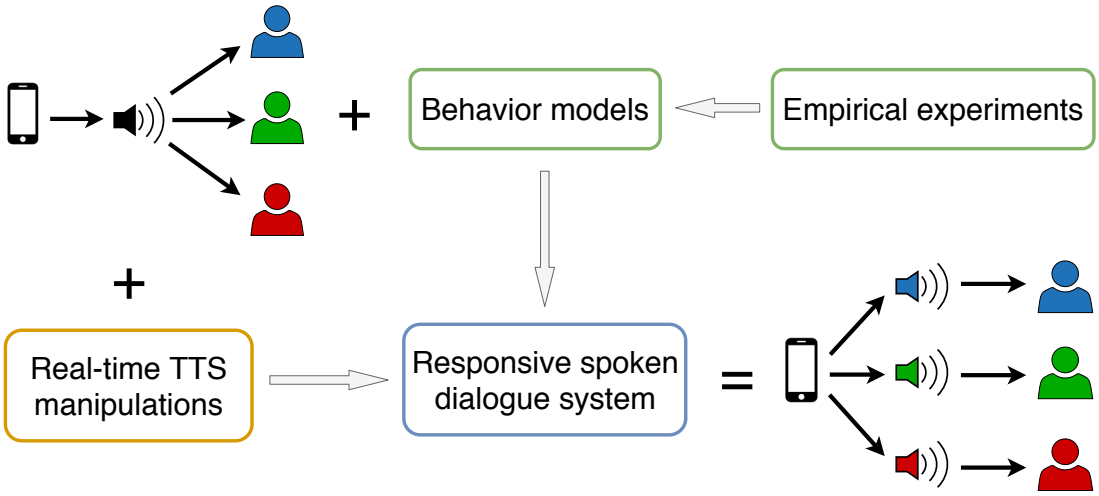


Figure 3.3: A suggested roadmap from static outputs (top left) to personalized outputs (bottom right) in SDSs (and see Figure 3.2). The green, orange, and blue blocks stand for the modeling, manipulation, and integration components described in the text, respectively. The '+' signs represent direct addition to a static system and the arrows go from components required as a feed to others.

the aforementioned notion of separating *what* the system says (determined by the NLG module) and *how* to say it (to be determined by the TTS with the help of this additional component).

This work addresses each of those facets, each with its own challenges that require a profound work to investigate. For a SDS to accommodate its speech, it would not only need to support dynamic, on-demand changes of its TTS component's output, but would also need its ASR component to be able to identify and track specific features in the user's speech to update its representation of those features. Completing the cycle, these representations can then be used as additional input for the TTS components to determine how those would influence the system's speech output. This process is individual for each speaker and may occur over long periods or multiple interactions, depending on the desired degree and characteristics of accommodation. Furthermore, this process may involve other components of the SDS as well. For instance, the DM might consider changes in the user's speech when making decisions, e.g., based on apparent mood and calmness. The NLG component could make alterations to its output to better fit the vocal changes of the system and the user's dynamic state, such as shortening sentences and omit additional information if phonetic indicators of hurry or urgency are detected

in the speech input (Edworthy et al., 2003).

3.3.3 Accommodation levels – terminology

In addition to all the above, another design choice for accommodative SDSs, which is not explicitly required in human speech, concerns the overall *level of accommodation* the system introduces. This determines the fundamental behavior and form of variation in the system's speech, regardless of specific phonetic features, utterance contents, etc. The variation levels (or properties) described below can potentially be combined in different ways to achieve the desired system behavior. They are, at least to some extent, analogous to the aforementioned processes conducted by humans in social interactions, with the key difference that humans don't need to defined and think about them separately, if at all. The utilization of these levels is demonstrated in the context of phonetic accommodation, which is one case of dynamic change of speech (Beňuš, 2014; Schweitzer and Walsh, 2016; Weise et al., 2019). Chapter 10 further discusses and demonstrates the integration of such properties into a SDS.

Different terms are used in the literature to describe systems that can change their output. This often leads to inconsistencies and mix-ups in terminology. Definitions of five core properties of accommodative TTS that should, once accomplished, grant a more dynamic appearance, along with their suggested use and potential fusion with one another, are suggested below. These terms seek to distinguish between the different capabilities that can be integrated into SDSs and the way they relate to humans' behaviors and each other.

Adaptive – the vocal behavior evolves between and/or during interactions.

This property refers to the system's use of any mean to *dynamically* change its speech-related behavior, regardless of the source of influence or specific realizations. That means, for example, modifying the base behavior, extending the variability, or reflecting the system's changes earlier. This can happen between interactions or within a single, usually longer, interaction. The former could be more useful for C&C systems or task-oriented SDSs that are used by many users, while the latter is more suitable for social systems like chatbots and PAs (see Section 3.2). Ultimately, this is a means for the system to improve its performance and accessibility based on previous interactions. However, for some applications, like a computer-assisted pronunciation training (CAPT) system, for example, it would be better to "reset"

their behavior between each use to offer a better experience.

Flexible – on-demand speech manipulation *without retraining the voice*.

This property refers to the technical capability to alter the system's speech output on request, which is achieved either via modifications in the voice's representation and parameters or by manipulating the outputted waveform directly using signal processing methods. Note that this does not entail the way this capability is used, and especially not that it is applied automatically. Moreover, as mentioned above, the technical capability to control the voice alone is not enough to create an accommodative behavior. This would also require additional data to be transferred to the TTS component to determine what manipulation to perform. To that end, the modeling steps can be built on top of this technical ground. It is important to note that this property compensates for the inherent ability in humans to control their voice at will, and therefore does not directly represent any specific element of humans' speech behavior like the other properties.

Responsive – changes are influenced by some *external* speech input.

A responsive system can, for instance, detect some target features in the user's speech input and, after comparing them to the system's representation of these features, guide the TTS on how to update them (typically, to make them more similar to the user's input). This requires a model that simulates these steps (as the one in Chapter 7). Yet this model would not have an independently defined behavior and it could only directly become more similar (or dissimilar) to the user in some fashion. Such models can be designed to imitate the user's immediate output from previous turns (like in Levitan et al., 2016) or to gradually match it based on some parameters like sensitivity and interaction's history, as demonstrated in Raveh et al. (2017b). This property represents the idea that humans change their speech (and behavior in general) when interacting with other people, which is a key aspect of the communication accommodation theory (CAT).

Characterized (Profiled) – the system's voice has its own base behavior.

Giving a "character" (or a *profile*) to a voice means that it has a specified base behavior, which might include general properties of accommodation. This can also be seen as a *role* the system plays in a conversation, e.g., if built based on a certain HHI scenario (Silber-Varod et al., 2018). In that case, the system would try to

stick to some pre-defined model, which contains the required information to fulfill said role. A system might also have several profiles to switch between based on the settings and goals of the interaction. In the context of accommodation, that would include, for example, the degree of accommodation, its timing, or how strongly the system will try to influence the user's speech. This property represents the fact that a human being has a certain – however complex – personality. More specifically, a vocal identity, which will be expressed in spoken interactions. This idea is the basis for the simulations presented in Chapter 8, where each generative model can be seen as a core behavior.

Variable – variations on top of the base behavior are yielded.

Some variations can be introduced based on the base profile. These are relatively minor differences that deviate from the voice's characteristics or enhance them in some way. From a system's point of view, the main purpose of such variations is to make the output style non-deterministic and therefore less repetitive and predictable. From the human point of view, this coincides with the difficulty to reproduce identical utterances in exactly the same way every time. Moreover, people, though having their individual personality, would speak differently based on various factors outside a conversation like their mood, the environmental conditions, time constraints, etc. This property comes, therefore, to grant some smaller-scale dynamics to the voice, in particular when its base behavior is deterministic. Chapter 8 explains in details an approach to achieve such variational behaviors.

This work concentrates on facets concerning the properties *responsive* (Chapter 7), *characterized*, and *variable* (Chapter 8), which have been given little attention in SDS research. Together, these properties result in a flexible, non-deterministic output derived from a defined core behavior, which might vary. Added to the responsive output of a system, this creates a system that adapts to the user according to its own base behavior and probabilistic variations.

3.3.4 Systems with accommodation capabilities

With the advancement of speech and NLP technologies, the development of SDSs has been accelerated in recent years. These technologies are closing the gaps between computer and human speech quality and therefore help to investigate vocal accommodation

in HCI with systems that can more closely simulate human performance. Particularly, the overall quality and dynamic manipulation of modern TTS methods make it possible to change certain aspects in a system's voice (see Section 9.2.2 for further details and examples). SDSs with vocal accommodation capabilities typically focus on a few phonetic features at most, and sometimes only a single one. Even though this does not provide a comprehensive coverage of all vocal changes that can be introduced in a conversation, the method of tracking and adapting the one feature can often be applied to others. An overview of the development process of adaptive SDSs and relevant methods is given by Bernsen et al. (1998) and Levitan (2020).

Suzuki et al. (2003) introduce a system with pure prosodic vocal capabilities with no lexical content. In an experiment, it communicated with the participants used hummed sounds with different fundamental frequency (f_0), intensity, and "speech" rate values. The system was adapting to the participants to different degrees over the several interactions, after which the participants rated the system on different communication aspects, like cooperation and friendliness. It was found that interaction series in which the system converged to the users with some degree of independence rather than cases where the system directly changed toward the user or completely mimicked the user's vocal behavior. This is an important insight, since this is also not likely to happen in HHI, as each speaker has a different general speaking style which varies differently if and when accommodating to an interlocutor (and see definitions in Section 3.3.3). Another approach was taken by Acosta and Ward (2011), whose system explicitly used mimicking for conveying emotions. The assumption was that if participants use certain vocal characteristics to convey specific emotions, they would also perceive the same emotions from an interlocutor's speech with the same characteristics. Lubold et al. (2015) investigated pitch accommodation in HCI with a system that supports different accommodation patterns. One of these patterns was used while participants interacted with the system over multiple short tutoring sessions. Ratings of perceived rapport showed that the method using pitch shifting of the system's pitch towards the user while keeping the original intonation was most successful. Like before, these results are in line with the approach of converging with respect to some concrete features while retaining more general behavioral patterns. Tackling vocal accommodation from a different direction, Ward and Nakagawa (2004) developed a system that predicts the speech rate of phone operators based on their interlocutors' speech. While the method was somewhat simplistic and

was applied to monotonic conversations with predictable content, it nonetheless shows that some general tendencies can be found based on vocal behavior in a conversation. As for implementation and integration, Levitan et al. (2016) use a modular way for adding accommodation capabilities to a SDS while separating the technical methods from the modeling details. This approach is similar to the one used in this work, with a major difference that here temporal aspects and non-linear methods were used to measure and analyze accommodation. While the system in Levitan et al. (2016) always directly converges towards the user at a defined rate, the modeling in this work aims to give the system an individual way to *respond* to the user's behavior based on some core profile, as explained in Section 3.3.3 and Figure 8.9.

II

Experiments

Chapter 4

Vocal Accommodation in Real-World Sales Calls

Vocal accommodation occurs in spontaneous human-human interaction. Its effects are investigated in this chapter using a collection of authentic sales calls. As speaking is an essential part of sales representatives' everyday work, controlling their voice is key for success. In addition to the extent of the effects, it is also examined which of the speakers leads the accommodation process in succussful and failed calls.

4.1 Harnessing speech alignment in conversation intelligence

Sales managers have always been impelled to be able to reason why some of their representatives (henceforth *reps* or *account executives (AEs)*) consistently attain and even exceed their goals while others do not (Kovac and Frick, 2017). Yet, sales executives rely on data that are inherently flawed, as it is based on reports from sources like customer relations management (CRM) systems. Such systems contain only "dry" details about deals' stage, along with high-level numeric data and some, typically subjective, estimations regarding their potential in the following stages. That leaves the executives in the dark regarding the happenings at the front lines and the small-scale, day-to-day conducts and modi operandi, which are critical for AEs' success. As a result, the reasons for losing or winning a deal often remain a riddle, making sales seem more like art based on anecdotes rather than a scientifically explainable practice (Yohn, 2016; Martin, 2017). In his seminal book, Gladwell (2006) states that "Part of what it means to have a persuasive personality is that you can draw others into your own rhythms and dictate the terms of the interaction" (p. 83). Supporting that, Orlob (2018) found that star reps[7] are able to make prospects increase their speaking rate to match theirs, bringing the two sides closer with respect to this speech property. However, the scientific analysis is far from their everyday work, and such phenomena are passed on as general tips and are not taught or explained in detail to the AEs. Conversation intelligence (C-IQ or CI) systems aim to bridge over this gap and connect the scientific side of sales to the field to help reps to improve their performance using measured, interpretable methods. Being part of verbal interaction, vocal accommodation is a facet that can shed light on certain processes occurring during a sales call. Indeed, this phenomenon has been given attention in analyses and evaluations of AEs. It was found, for example, that distinguished sales reps let the prospects talk more and keep certain parts of the calls shorter than low-performing reps (Orlob, 2017a).

Some phonetic features, such as Fundamental frequency (f_0), can be interpreted

[7]Sales representative whose performances (and specifically their closed-deals rate) are exceptionally high are often referred to as *star reps*. The specific criteria typically include a wide range of behavioral and business metrics, but those are defined internally per company and do not have a common absolute definition.

and explained by people with no phonetic training, like AEs. Furthermore, speakers can easily control it, making it a tangible tool for the AEs to exploit in their sales calls. This, as opposed to more complex acoustic features like mel-frequency cepstral coefficients (MFCCs) or changes in the long-term average spectrum (LTAS) that are often used in phonetic accommodation studies (e.g., Levitan and Hirschberg, 2011; Borrie et al., 2019) and are hard to directly control during a conversation. For this reason, the analyses presented in this chapter focus on f_0, although other suprasegmental features showed similar effects. To that end, a large-scale corpus of real-world sales conversations was collected (Section 4.2). Beside the size advantage, the use of such a corpus takes the study of vocal accommodation out from the controlled laboratory environment into the wild for a practical goal. The conversations in this corpus are therefore highly flexible and generally structure-free, unlike lab experiments like those presented in Chapter 5. It is also important to note that beside the overall goal of closing a good deal, there are no specific instructions given to the speakers on both sides regarding how to speak or what to say. By extension, they are also more authentic and spontaneous, since the interlocutors are driven solely by their own behavior and motivation to succeed and are not given an artificial temporary role. Cross-recurrence quantification analysis (CRQA) (Zbilut et al., 1998), a bivariate correlation technique, was used for the analysis (Section 4.3). This method finds instances where coordinates of two time series occur close to each other within a certain radius in a phase-space continuum.only Since CRQA evaluates the degree to which the similarity of two time series changes over time and can also determine the leading relationship between them, it is suitable and informative for analyzing accommodation. The contributions of this study are therefore both in the methodology for measuring accommodation and its practical application in a real-world scenario.

4.1.1 Conversation intelligence

The way people communicate and behave in inter-person situation influences the manner in which conversations unfold. These influences can be analyzed and interpreted to uncover conversation-level trends, which may differ from the linear turn-by-turn changes. Conversation intelligence (C-IQ or CI) is a relatively newly coined term and a field of research that flourished due to advancements in neuroscience, communication science, and machine learning. It complements other types of human intelligence, like *emotional*

intelligence (c.f. *Theory of Multiple Intelligences* (Gardner, 1983; Davis et al., 2011)), as the ensemble of conversational aspects in human communication *beyond the surface words and shared information*. Glaser (2016) explains and demonstrates how C-IQ can be learned and improved, with emphasis on the ability to gain trust and maintain more successful communication. Although C-IQ comprises many further aspects that are beyond the scope of this work, the overarching idea that communication quality is a defined, learnable skill that can be refined is highly relevant for vocal accommodation but has not been explored in previous work.

Narrowing down this broad idea, Silber-Varod (2018) discusses how conversations can be *managed*. This includes both the structure and evolution of a conversation over time and the dynamics between the interlocutors' based on their roles in it. Noticeably, it is concluded that some speech-related phenomena in conversations tend to be more complex and unpredictable than they seem in their surface form. One of those phenomena is phonetic entrainment, which was found in long-term influences of the speakers on each other. Such works dealing with managing spoken interactions become even more relevant with the growing interest in intelligent conversational systems for personal usage (Mehr, 2017). Recently, the importance of C-IQ has pervaded the enterprise sector and created new businesses. Two main motivations were the utilization of computer-based customer services (Gnewuch et al., 2017) and the use of conversation intelligence services for inside sales calls to train AEs and improve their performance (Orlob, 2017a,b).

4.1.2 Inside sales

In recent years, many companies have adopted the concept of *inside sales*, where business-to-business (B2B) sales are done using web-based conferencing solutions, as opposed to face-to-face meetings with the clients. Recent technological advancements allow automatic recording and transcription of those inside sales calls and aggregation of large-scale datasets. These datasets include the audio of the calls and sometimes annotations such as the speaker turns and performance rating of the AE. These conversations have a typical process: First, a sales development representative (SDR) reaches out to a potential client (the *prospect*) who has expressed interest in the company's product, which initiates a *lead*. Subsequently, the SDR shares basic details about the product and how it can help the prospect. Finally, if the SDR has managed to elicit initial interest, the lead turns into an *opportunity*, and a demo call with a sales representative is scheduled.

Such demo calls are often done using a web conferencing tool, such as Zoom[8] or GoToMeeting[9], which allows both sides to share their webcams and screens. A collection of such initial opportunity calls is analyzed in here (Section 4.2). It is known in the B2B industry that since these calls are the first personal contact, the behavior and verbal skills of the AE have a large weight in the success of the call.

4.1.3 Influence of speaker roles

Although accommodation is not a traditional measure in C-IQ, there have been attempts to use it as an additional factor in conversation evaluations. Glaser (2016) emphasizes the importance of the turn-level alignment between speakers in a business call. Lack of alignment may result in a skeptic and even resisting behavior that might lower the success chances of the call (see Table 2 in Glaser and Tartell, 2014). At the very least, this demonstrates effect of interlocutors being attentive not only to the content of the conversation, but also to the way it is delivered. In B2B calls, this is especially relevant for the AEs. Silber-Varod (2018) shows how convergence effects indicate power relations facets of C-IQ analyses. Specifically, speaker "dominancy" is often hard to spot on the surface, but becomes more prominent when examining the vocal relations between the speakers. Furthermore, Abrego-Collier et al. (2011) discuss how the judgment of a speaker's role in a conversation influences the phonetic productions and vice versa. Analyses like these improve the understanding of vocal behaviors in human-human interaction (HHI). The idea of speakers' behavior being modified due to their role in the conversation and their perception of the other interlocutor is a key concept for the presented study.

4.2 Dataset and feature extraction

A collection of real-world calls with similar characteristics to those described in Section 4.1.2 was used in the study presented here. These calls were all made by trained sales representatives and were aimed at high-stakes deals[10]. To make the collection

[8] https://zoom.us

[9] https://www.gotomeeting.com

[10] In this case, around US$100,000 each, as opposed to occasional mass calls to random people for selling small products where the stakes and risks are very small in comparison. The reps know that their

more homogeneous, calls of a single sales company were selected. Another constraint was that the collection comprised only calls from a very early stage of the sales opportunity (see Section 4.1.2) that is also the first encounter between the participating AE and the prospect. Therefore, the observed behavior patterns in these conversation are not influenced by the previous verbal interactions between the two speakers. the structure of these calls typically includes an overview of the prospect's business, followed by a deeper explanation of the sold product by the salesperson, and typically further negotiations. It is important to note that although professional AEs prepare for these calls as part of the daily work, they are still spontaneous and are in no way scripted. The calls in this collection were conducted using the Zoom video conferencing platform and were recorded automatically – without any intervention from either side – by an external conversation intelligence service. The calls were transcribed and diarized using the internal automatic speech recognition (ASR) system of the service. Participants were notified of the recording, in compliance with all relevant laws and rights.

In total, 708 calls were analyzed, spanning more than 442 hours (mean 37.5 ±15 minutes). Furthermore, only calls longer than 15 minutes were selected, as shorter calls are often unsuccessful connection attempts or brief updates that are not representative of the desired conversation structure and dynamics. A single AE and a single prospect participated in each call. Call recording started immediately following the first greet from the prospect's side, and stopped when the AE terminated it. This eliminates segments during which one party is waiting for the other to join, and keeps only segments where both parties are present. Each prospect only ever spoke with one of the 26 AEs (12 female). Both interlocutors in all the conversation were native speakers of American English and worked in an English-speaking company. For the purpose of this study, a call was defined as successful if a follow-up call under a more advanced stage was initiated or when an advancement in the opportunity was marked within one month. These criteria follow common success measure conventions of B2B calls[11]. This resulted in 51 calls (7.2 %) being defined as successful, which is within the industry-standard

performances are evaluated, pushing them to do their best in every single call.

[11]The benchmarks for successful deals are much more elaborate in practice and usually consider an entire selling process that comprises a series of calls. However, many of those criteria are not relevant or cannot be enforced in the scope of this study.

ratio for such early-stage calls.

To increase temporal resolution, the audio signals were split into two-second slices (cf. Section 6.3). This increases the number of datapoints the CRQA considers. Splitting the turns also creates equal, consecutive, continuous time units in the conversations, which are more comparable, without introducing artificial boundaries by dividing it into a pre-defined number of parts based on some arbitrary criterion. The slicing was done per turn, so that a slice contains only the speech of a single speaker. Any remainder of a turn got a slice of its own. When a speaker was not speaking (e.g., during the turn of the other interlocutor), it was assumed that the last produced value was maintained until the speech is renewed and a new value can be measured. This way, no discontinuities are created and the same number of datapoints can be extracted for both speakers to create a better temporal representation of the conversation. Feature extraction was done using the system described in Chapter 10 (and see Raveh et al., 2018). The values were measured using Praat (Boersma, 2018) scripts that extracted the f_0 value from the middle of each slice. Afterward, the list of measures was turned into two equally long time series: First, the values for each speaker were separated into two lists. Subsequently, missing values, e.g., due to non-speech slice, were replaced by the most recent valid value of their respective speaker, as motivated above. Finally, if there were missing values at the beginning of a list, the first valid value of that list was used, under the assumption that the first utterance represents the immediate time prior to it. The resulting two lists had the same number of values representing the same timestamps along a single conversation, and were used as the input time series for the CRQA, which can be assumed to be non-seasonal and non-stationary. This process was performed independently for each conversation.

4.3 Cross-recurrence quantification analysis for measuring vocal accommodation in conversations

4.3.1 Capturing accommodation with CRQA

Depending on the circumstances, HHI may involve different communication channels, such as facial expression, hand gestures, and eye gaze. The analyses here concentrate on the phonetic level, as it is the primary modality used for conveying information in sales calls, even when video or screen sharing functionalities are available as well. It has been

shown in studies based on communication accommodation theory (CAT) (Section 2.1) that mutual vocal adjustments in HHI increase the success rates of conversations (Pickering and Garrod, 2004) and affects the social distance between speakers (Schweitzer et al., 2017a). Accordingly, effects of the same nature have been found in B2B sales calls as well (Orlob, 2019).

Linear methods for measuring accommodation rely on the chronological, turn-by-turn order of the interaction. As explained in Section 2.2.2 and Figure 2.2, these methods are limited to the detection of local effects that evolve gradually across adjacent turns. Non-linear methods, on the other hand, do not rely on turn adjacency and can find long-term relations between the speakers' speech productions on the conversation level. For instance, accommodation may occurred at some point in the beginning and be continued at a later time. This is especially useful for long interaction, like the sales calls in the corpus used here, where the insights from more general view are useful for improving performance. This encourages treating the interactions as continuous event rather discrete parts, and opens a variety of time series analysis methods. CRQA is one such method, which offers more than the direct comparison of large pre-defined chunks of neighboring turns, as is typically done in accommodation studies in spontaneous conversations (e.g., Levitan, 2013; Rahimi and Litman, 2018). It utilizes phase-space embedding, which describes the temporal evolution of trajectories of a dynamic system by projecting their embedding onto some common space.

An overview of the CRQA method is given by Wallot and Leonardi (2018). At its core, this method compares delayed instances of the phase-space trajectories of two time series. This allows for finding more general patterns in the time series characteristics and how they interact. It is especially suitable for studying accommodation and related phenomena, as it detects times in which the time series (here, the speakers' productions) are similar. Moreover, it can mathematically show which of the time series lead the alignments or whether the changes were done in synchrony. This is useful for describing different the types of accommodation described in Section 2.1.1, like synchronicity and alignment. CRQA has already been used in HHI research. For instance, Duran and Fusaroli (2017) used it for analyzing and predicting speech differences in scenarios with disagreement and deception tasks. A similar method was used by (Borrie et al., 2019) to measure conversational entrainment for assessing speech pathology. It was found that sessions with longer periods of synchronization were rated as more successful by

therapists. This is a good example of a cooperative interaction with a common goal that accommodation contributes to its success. In sum, CRQA can be used to objectively quantify and describe accommodation between speakers dynamically across entire conversations. Sections 4.3.2 and 4.3.3 explain the technicalities of CRQA, how its output can be interpreted, and how it is used in the study presented in this chapter.

4.3.2 Recurrence detection

Recurrence quantification analysis (RQA) is a method for non-linear data analysis that quantifies the number and duration of recurrences within a dynamical system presented by its state-space trajectory, which is typically the realization of a sampled time series. It was introduced by Zbilut and Webber Jr (1992) and later extended by Marwan et al. (2002) and Webber Jr and Zbilut (2005). A *recurrence* (also, *re-visitation*) is a time in which the trajectory returns to a state it has visited before. Recurrence can, therefore, be defined as the binary function

$$R_{i,j} = \begin{cases} 1, & \text{if} \quad \|\vec{x}(i) - \vec{x}(j)\|_d \leq \varepsilon \\ 0, & \text{otherwise} \end{cases}, \tag{4.3.1}$$

where i and j are samples of the time series, d is the number of embedding dimensions, and ε is the threshold radius distance below which two cross-trajectory points are considered similar, as explained in Section 4.3.3. CRQA is an extension of RQA for analyzing recurrence quantification between *two different* time series rather than a single one. As such, CRQA is a quantification technique for non-linear data analysis that describes when and to what extent concurrences (or *co-visitations*) occur in the two time series. These quantification techniques are based on *recurrence plots* that for each pair of samples i and j from the two time series show the times at which a phase space trajectory is similar, i.e., when $TS_i \approx TS_j$ (and see Figure 4.1). The recurrent points $R_{i,j}$ are colored if their value is 1 or remain unmarked otherwise. The main diagonal of the plot is called the line of synchrony (LoS). A high number of recurrences along this line indicates synchrony between the time series. However, diagonal recurrence lines can be formed above and below the LoS. Such diagonals, especially longer ones, represent delayed (lagged) synchrony between the time series and can be used as an assessment of similarity between the processes. In the context of accommodation, these diagonals imply accommodative

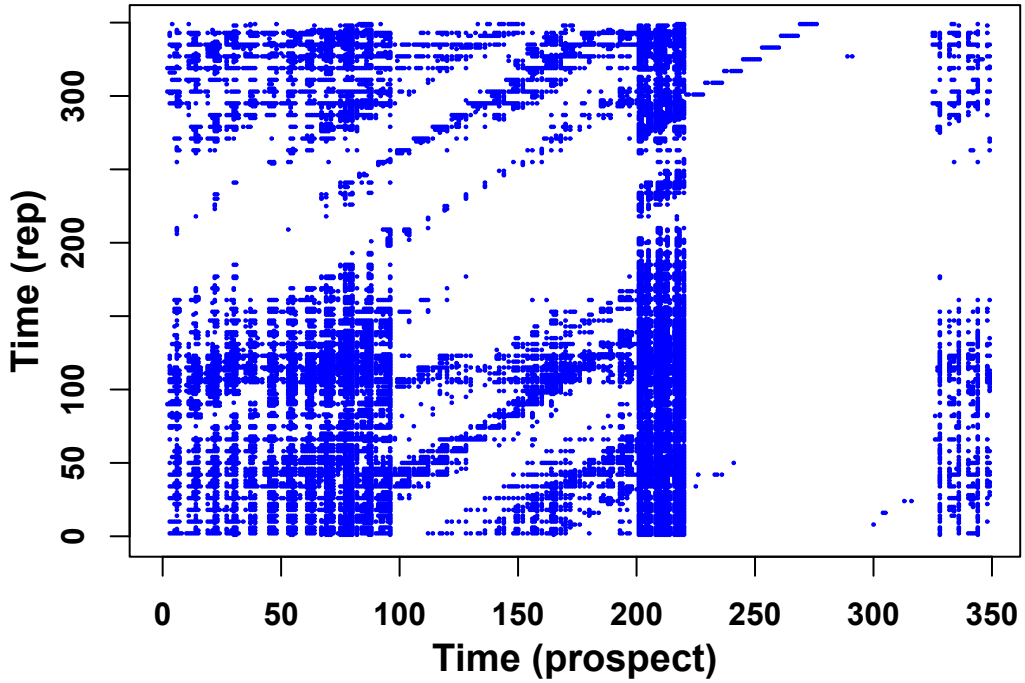


Figure 4.1: A recurrence plot generated for one of the analyzed conversations. The y-axis marks the conversation timeline, in slices, of the AE, and the x-axis of the prospect. Each blue dot represents a co-visitation of a similar state. Blue dots forming a diagonal line indicate sustained recurrence between the two speakers (see description of NRLINE in Section 4.4.1 for details). Note that the timestamps on the axes are not the slices, but the embedded call time. For example, the diagonal structures between timestamps 100 and 200 of the x-axis show such lasting recurrence. Diagonal lines above the line of synchrony (LoS; the central diagonal line) indicate that the speaker on the y-axis leads the x-axis, and vice versa for lines below the line of synchrony (LoS). The blank area between timestamps 220 and 330 of the x-axis point to a portion of the call where the speakers were more distant from each other.

processes led by one of the interlocutors. If the diagonal stretches above the LoS, the speaker plotted on the x-axis leads the accommodation and vice versa. The closer the diagonal is from the LoS, the faster the process occurred, i.e., the led speaker aligned his behavior to the leading speaker after a shorter time. This ability to not only detect latent accommodation but also determine its initiator on a fine-grained time scale enables the description and detection of more complex accommodation behavior, such as delayed synchrony (see Figure 2.2).

4.3.3 Parameter tuning

CRQA has three parameters:

1. **Delay** – estimates the temporal shift required to make the two time series maximally correlated. It is measured by the same time unit as the time series (here, two-second slices; see Section 4.2).

2. **Embedding dimensions** – are the number of dimensions into which the datapoints are embedded. These dimensions are delayed copies of the original time series TS_t created by adding a lag k to them. Typically, multiple n lags are considered, which create the dimensions of embedding TS_{t+nk}.

3. **Radius** – determines the margin within which two datapoints constitute a recurrent instance. Distances between the datapoints are measured in the embedded space defined by embedding dimensions, using the same unit used for measuring the values of the time series.

These parameters are a key aspect in CRQA, and how they are set is decisive for its outcome. However, although some best-practice guidelines exist, like those suggested by Coco and Dale (2014), there is nevertheless no standard way for optimizing these parameters and their determination depend on the nature and characteristics of the data. An optimization method similar to the one presented by Marwan et al. (2007) was utilized here. The average mutual information (AMI) of the time series' shifted instances is defined as

$$AMI\left(TS_{i(t)}, TS_{j(t+\tau)}\right) = \sum_{i,j} p_{ij}(\tau) \log\left(\frac{p_{ij}(\tau)}{p_i p_j}\right), \tag{4.3.2}$$

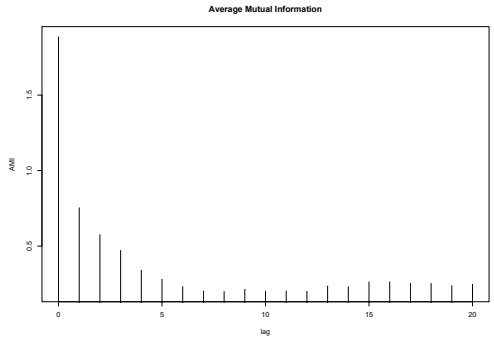


Figure 4.2: The average mutual information of the time series values as a function of the lags considered. The x-axis shows the considered lags and the y-axis the mutual information index (AMI) in bits.

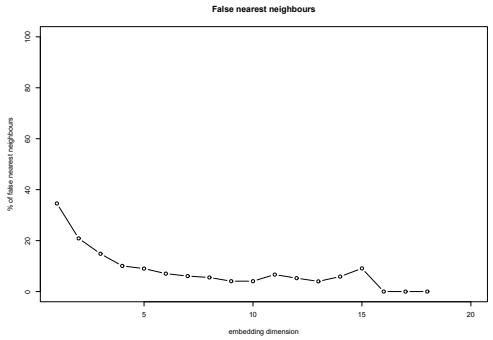


Figure 4.3: False nearest neighbors percentage as a function of the number of embedded dimensions. The x-axis show the considered numbers of embedded dimensions and the y-axis the percentage of false nearest neighbors.

where i and j are values from the two time series, t is the original starting time of the time series, τ is the amount of shift between the time series, and p_{ij} is the probability that $R_{i,j} = 1$. Note that only the shift τ influences this value and not the absolute initial time t. The delay parameter was subsequently determined by finding the lag value τ that minimizes the average mutual information between the two time series, as follows:

The lag with the *lowest* average mutual information was selected, regardless of whether and when the values started to level off. This provides a delay that is not too short to miss the mutual differences, but also not too long to lose the dependency between the time series. The number of embedding dimensions was obtained using false nearest neighbors (Kennel et al., 1992). This algorithm determines the minimum embedding dimension necessary to reconstruct the state space of a dynamical system with time delay embedding, as explained by Abarbanel and Kennel (1993). A neighborhood diameter equal to the standard deviation of the time series was used, and a limit of 20 embedding dimensions (which was never reached) was set. Figures 4.2 and 4.3 show examples of mutual information and false nearest neighbors optimizations using the data used for the study presented here.

$$\hat{\tau} = \underset{\tau}{argmin}\left(AMI\left(t,\tau\right)\right). \tag{4.3.3}$$

Algorithm 1: CRQA radius optimization

Inputs : $\underline{desiredRR}$ – the desired approximated recurrence rate (RR) value
$\underline{optDelay}$ – the optimized delay
$\underline{optEmbedim}$ – the optimized embedded dimensions
$\underline{TS}$ – all values from both time series

Output: radius producing recurrence rate (RR) value closest to the defined desired recurrence rate (RR)

1 $n \leftarrow$ number of radius candidates
2 $R \leftarrow \{r_i, \ldots, r_n\} : r_1 = 0, r_n = \max(TS)$
3 $\forall r \in R : |r_i - r_{i-1}| = |r_{i+1} - r_i|$ `// evenly spread candidates`
4 $candR \leftarrow \varnothing$
5 **foreach** $r \in R$ **do**
6 $\quad currRR = crqa(ts_1, ts_2, optDelay, optEmbedim, \ldots^{12}).RR$ `// RR with candidate`
7 $\quad candR = candR \cup currRR$
8 $\quad$ **if** $currRR = 100$ **then**
9 $\quad\quad$ *break* `// recurrence rate (RR) cannot be higher than 100`
10 **end**
11 $optRadius = \underset{r \in R}{argmin}(|candR_r - desiredRR|)$

The higher n is (Line 1), the higher the chance of a recurrence rate (RR) close to the defined desired RR*. Also note that since recurrence rate (RR) represents percentage of values in the recurrence plot, 100 is its highest possible value. Therefore, and since the radii in R are traversed in increasing order, the search stops if this value is achieved (Line 9). As explained in the text, in the presented studies* $n = 20$ *and* $desiredRR = 10$ *(see Coco and Dale, 2014).*

Following that, the radius was calculated in two steps (Algorithm 1). First, the goal recurrence rate (RR) and a list of potential radii were initialized. Here, the goal RR was set to 10 %, which is considerably higher than in similar studies, e.g., 2 % to 5 % in Coco and Dale (2014). Setting the goal RR to a higher value results in a stricter optimization that includes only (even) closer recurrences. The potential radii were generated by evenly spreading 20 candidates from 0 to the maximum value in the time series. Then, each radius, along with the already optimized delay and embedding dimensions, was used to perform a CRQA. A stricter policy was introduced in this step

[12] As calculated by the CRQA package for R (Coco and Dale, 2014) with the additional arguments (unlisted in the algorithm itself) *rescale=0, normalize=0, minvertline=length(ts1) / 100, mindiagline=length(ts1) / 100,* and *tw=0.*

as well, as only recurrence lines longer than 1 % of the longer time series' length were counted, as opposed to the typical setup that considers lines of any length (e.g., as in Borrie et al., 2019). With an average length of 37.5 minutes, this means that only lines longer than 11 time units (22 second) were considered. This ensured that only long-term recurrences were taken into account, and shorter, possibly more random effects, were filtered out. Finally, the candidate radius that resulted in the smallest absolute distance from the goal RR was chosen to perform the CRQA. This process is summarized in Algorithm 1. The optimization processes of all three parameters were done separately for each conversation.

4.4 Analysis

Time series-based analysis methods like CRQA can be used in many ways. The measures used in this study are explained in Section 4.4.1, followed by the results they yielded for the sales calls collection in Section 4.4.2.

4.4.1 CRQA output values

Various measures can be computed based on a recurrence plot[13]. Some deal with the LoS and other diagonals across the recurrence plot, while others consider vertical and horizontal lines. Below are the definitions of the measures used in this study. For simplicity, it is assumed that the time series lengths are equal, so that $N_i = N_j = N$.

Recurrence rate (RR) – The percentage of recurrent points in the plot, i.e., the percentage of similar values between the two time series out of all the values as calculated using the parameters described in Section 4.3.3. It is defined as

$$RR = \frac{1}{N^2} \sum_{i=1}^{N_i} \sum_{j=1}^{N_j} R_{i,j}. \tag{4.4.1}$$

This number corresponds to the amount of colored points in the recurrence plot. The lengths N_i and N_j of the time series are often – but not necessarily – equal

[13] See detailed overview in Marwan et al. (2007) and summary with additional measures and examples in http://www.recurrence-plot.tk/rqa.php.

Determinism (DET) – The percentage of recurrences forming diagonal lines in the recurrence plot given a minimal length threshold $l_{\min}$. It is calculated as

$$DET = \frac{\sum_{l=l_{\min}}^{N} lP(l)}{\sum_{l=1}^{N} lP(l)}, \tag{4.4.2}$$

where l is the length of a line and $P(l)$ is the length histogram of all diagonal lines. Note that $l = 1$ refers to lines of length 1, i.e., a single recurrence point. Similarly, $l = 2$ are lines spanning over two timestamps (shortest lines possible). Short lines lengths are very forgiving in the case of accommodation, for they can be formed abundantly and therefore not necessarily indicate a meaningful accommodation process. For this reason, $l_{\min}$ was set to $\frac{N}{100}$ in this analysis, as detailed in Section 4.3.3.

Another measure, Laminarity (LAM), can be calculated the same way, but for the vertical lines in the plot. In that case, a $v_{\min}$ is defined and $P(v)$ provides the histogram of vertical line lengths.

Number of lines (NRLINE) – The total number of lines N_l formed in the recurrence plot per the definition of the DET measure. In the context of accommodation, this is the number of accommodation "instances" between the two speakers lasting at least $l_{\min}$ time units. Also referred to as *sustained recurrence* by Borrie et al. (2019), this measure not only shows the number of detected accommodation effects, but also how long they last.

Average length (L) – the average length of the diagonal lines, i.e., the average total accommodation time between the speakers. A higher value indicates longer average individual accommodation timespans in the conversation. It is calculated as

$$L = \frac{\sum_{l=l_{\min}}^{N} lP(l)}{\sum_{l=l_{\min}}^{N} P(l)}. \tag{4.4.3}$$

The average length of vertical lines, trapping time (TT), is achieved in a similar way when using the same formula for the vertical line length and the histogram of vertical line lengths.

Maximal length (maxL) – The longest diagonal line in the plot (excluding the LoS in RQA). This is the longest, uninterrupted timespan over which accommodation

between the speakers has lasted. It is determined by

$$L_{max} = \max_{l}(\{l_i;\ i = 1, \ldots, N_l\}). \tag{4.4.4}$$

Similarly, the longest vertical line can be determined by traversing the vertical lines.

Entropy (ENTR) – The Shannon entropy of the probability distribution of the diagonal line lengths longer than the minimum length $l_{\min}$. Describes the variability of the amount of accommodation instances across the conversation. Higher value indicates more varied lengths. Based on Shannon's entropy formula, this value is defined as

$$ENTR = -\sum_{l=l_{min}}^{N} p(l) \ln p(l), \tag{4.4.5}$$

where $p(l)$ is the probability of length l from the lengths probability distribution.

Normalized entropy (rENTR) – The entropy value normalized by the number of lines formed in the recurrent plot. This measure describes the length variation across multiple conversations and is therefore not too biased by special characteristics some conversations might have.

4.4.2 Results

Seven out of the output values described in Section 4.4.1 were measured for all 708 conversations and their distributions were checked for statistical significance. Since multiple variables were compared, the Bonferroni correction (Bonferroni, 1936) was applied, so that the overall error rate across all variables is $\alpha = 0.05$. Therefore, for a single comparison to be significant, its p-value must be lower than 0.007. The non-parametric two-sample Wilcoxon test (Wilcoxon, 1945) was used to determine the significance levels. Table 4.1 summarizes the means and p-values of these distributions for failed and successful calls. The RR mean value is about 10 in both groups, which matches the goal value set for optimization (Section 4.3.3). Significant differences between failed and successful calls were found for the values DET, NRLINE, and rENTR. The first two, along with their higher means for the failed calls, indicate more synchronous accommodation instances in the failed calls. The latter is harder to interpret, especially due to the similar means for successful and failed calls. It seems that the behavior in both cases

Table 4.1: P-values of the two-sample Wilcoxon test comparing the CRQA output values based on call success/fail and leading speaker along with their respective mean values. Significant values based on the adjusted p-value threshold are in bold.

	RR	**DET**	**NRLINE**	**maxL**	**L**	**ENTR**	**rENTR**
p (deal)	0.9	**$\ll$0.001**	**$\ll$0.001**	0.7	0.07	0.1	**0.0058**
mean success	10.1	2.2	84.0	31.3	17.8	1.4	0.8
mean fail	10.0	5.9	174.0	32.9	15.3	1.5	0.8
p (lead)	0.7	0.1	0.05	0.18	0.9	0.07	0.55
mean AEs	10.0	6.1	186.7	35.2	15.2	1.5	0.8
mean prospects	10.1	5.4	154.4	31.2	15.7	1.4	0.8

is equally predictable across calls, but differently. One possible factor influencing these behaviors is related to the interlocutor leading the accommodation effect, rather than the fact that it occurs at all, as shown below. The significant difference in the NRLINE measure stands in line with the findings of Borrie et al. (2019). The same tests were performed based on the leading speaker in each conversation (bottom part of Table 4.1), but no significant differences were found for this criterion.

To shed more light on the leading role, cross-correlation was used to determine which interlocutor led the change in behavior. Cross-correlation finds the degree to which two time series are synchronized with different lag values.. The correlation between the time series was calculated for each lag, and the value that made the series maximally correlated was selected. A positive value suggests that the first time series needs to be shifted forward to achieve maximal correlation and vice versa. In the context of accommodation, this cross-correlation indicates which speaker was leading the change. The cross-correlation function was not limited to a specific range of shifts, so that all possible lags were considered. The lag associated with the maximal correlation was used to determine the leader. Since the calls in the collections used here are from the same domain, it is also interesting to examine at which point of the conversation the maximal correlation occurs. Figure 4.4 shows the time, in percent, in which the maximal correlation occurred for AEs and prospects in successful and failed calls. The difference between the lag position distribution of reps and prospects is significant ($p = 0.0065$; $\alpha = 0.05$). However, no significant different was found between successful and failed calls for the same leading speaker. It is also evident that when the reps were leading, they

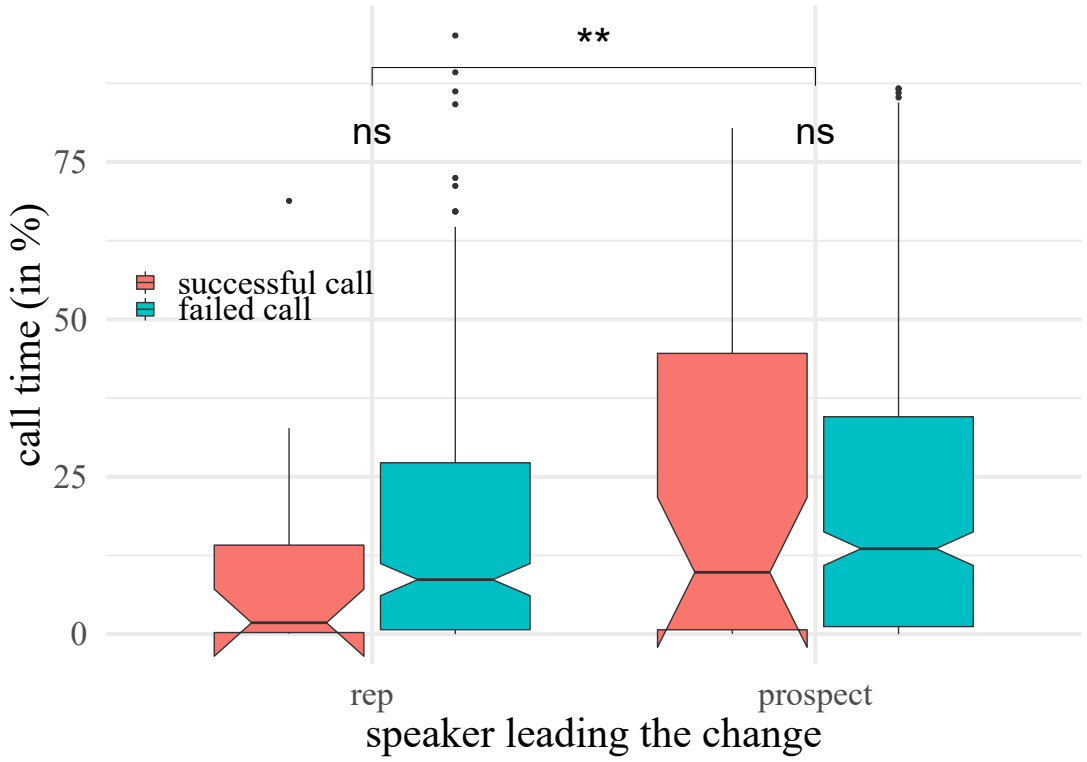


Figure 4.4: Comparison between the time of the call in which the maximal cross-correlation occurs. The x-axis groups the calls based on the speaker role, and the fill color further separates between successful and failed calls. The y-axis shows the conversation timeline in percent (50 % marks the middle of the conversation). The horizontal lines in the boxes represent the median and he notches stand for the 95 % confident level. The significance level calculated with Wilcoxon test comparing the two groups and their subgroups is given above the boxes.

consistently did so at an earlier stage of the conversation, all the more so in successful calls, whereas prospects' lead varied much more and generally happens at a much later time. Another known conversational element in sales calls is the floor time each speaker gets. As a general approach, AEs aim to let the prospect talk as much as possible. This is known to give them a better feeling during the call, and also give the reps more information and opportunities to understand what the customer wants to talk about. Indeed, a recommended practice is to track the *speech balance*, i.e., the ratio between the speaking time of the speakers. Besides speech balance, the frequency and timing of floor switching also provide some insight on the dynamics of the speakers' vocal behaviors. While each speaker should get a sufficient amount of time to talk, it is also important to

take and give the floor to the other interlocutor when necessary. Long monologues can make the listener lose concentration or lack of expression, which damages the interaction. Therefore, the *interactivity* of the speakers is important as well. While speech balance informs about the overall amount of time each speaker talked, interactivity complements this by informing how often a speaker gave the floor to the conversation partner. These two speech-related properties fit into the overall notion of vocal behavior. Speech balance was measured by

$$speech\_balance = 1 - \left| \frac{\sum_{\forall S \in S_A} dur(S) - \sum_{\forall S \in S_B} dur(S)}{\sum_{\forall S \in S_A \cup S_B} dur(S)} \right|, \tag{4.4.6}$$

where S_A and S_B are the slices in which speakers A and B speak, respectively, and the function *dur* returns the duration a slice. The yielded value between 0 and 1 indicates the percentage of the balance in terms of speech times, with 1 standing for "perfect balance", i.e., equal talking times for both speakers. As mentioned above, the overall speech balance only reveals part of the whole picture. Another part of it is the interactivity in the conversation, which is here defined as the percentage of slices in which floor change occurred after a sequence of longer than 1 slice was calculated. Sequences below this threshold were treated as backchanneling, which does not indicate speaker change. Interactivity and speech balance were measure for a superset of the dataset presented in Section 4.2, which consisted of more than 1,000 calls. Figure 4.5 shows the speech balance scores of successful and failed calls. On the one hand, it is clear that the lower the balance the more likely it was that reps had more floor time. The recommendation to avoid imbalance is reflected by the significant difference between balance distribution in successful and failed calls. On the other hand, prospects are only likely to talk more when the balance score is high, even more so in successful conversations. This is accentuated by the highly significant differences in both sub-groups. No significant influences of interactivity on call outcomes were found, and only a weak correlation between speech balance and interactivity was found ($\rho = 0.2$).

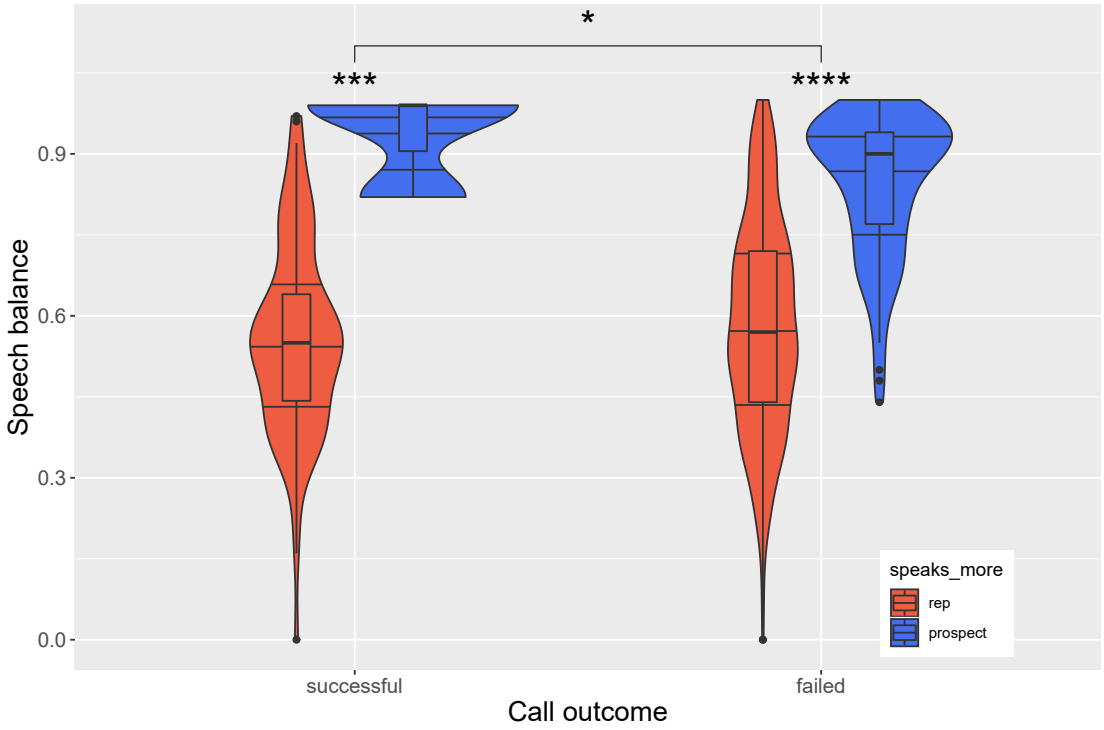


Figure 4.5: Comparison of speech balance distribution in successful and failed calls subgrouped by the speaker who had more floor time in individual calls. The width of the shapes represent the probability density. The inner boxes show the central quartiles of the data. The medians are marked by the thick horizontal lines in the boxes. The additional horizontal lines mark the 25 %, 50 %, and 75 % quartiles of the data. The asterisks above the shapes denote the significance levels of the comparisons between the main groups (successful vs. failed) and the subgroups (* $p < 0.05$, *** $p < 0.001$, **** $p < 0.0001$).

4.5 Conclusion

The study presented in this chapter investigated accommodation occurrences in real-world sales conversations using cross-recurrence quantification analysis (CRQA). Two main properties were examined, namely call success and role influence. The results show that successful and failed calls significantly differed in three of the CRQA output values. Although based on some HHI studies it might be hypothesized that recurrence is more likely to occur in successful calls, the means of two of the three values suggest the opposite. Yet, this stands in line with other studies from sales research that show more "desperate" behavior from the rep side when a deal is hard to close. This includes unconsciously showing assimilation towards the prospect and over-emphasizing details, with the hope that it will convince the prospect to close a deal (Orlob, 2019). However, these often achieve the opposite effect and are therefore discouraged in the sales industry. Another possible explanation is that reps give up the lead when a call is on the verge of failure and instead let the prospects lead to give them a better feeling. This, too, is a known effect in the sales business. On the other hand, utilizing cross-correlation lags proved to be useful for differentiating between the leader in the calls. When account executives (AEs) lead, they tend to do so at an earlier stage than prospects, all the more so in successful calls.

These findings suggest that AEs do not necessarily always lead the conversation, but know how and when to exploit this technique, consciously or not, to improve their stance in a call. It can be concluded that accommodation-related characteristics can be found in spontaneous, goal-oriented conversations and be used as a mean of analyzing their effectiveness. However, despite anecdotal recommendations and recommended best practices, it cannot be claimed that exploitation of these effects *directly influence* calls' success without further investigating, e.g., long-term performances of highly-rated reps.

Chapter 5

Shadowing in Sung Music and Human-Computer Interaction

After finding accommodation effects in human-human interactions, this chapter examines such effects in a more controlled setting. The studies presented here follow the shadowing paradigm and investigate accommodation in different facets of singing and human-computer interaction. The results of these studies lay the foundations for investigating accommodation in more conversational scenarios of human-computer interaction.

5.1 Shadowing paradigm

In a shadowing task, participants are instructed to provide vocal productions as a reaction to pre-determined stimuli. It is often used in empirical experiments (e.g., Goldinger, 1998) to examine how certain properties of these stimuli influence – or do not influence – participants' productions. A shadowing task is typically preceded by a baseline phase, where the participants provide the same productions without listening to the stimuli. The stimuli used in the shadowing phase can be determined based on the baseline production, to intentionally introduce a contrast between the stimuli and the participants' preference with respect to a certain feature, as done in Section 5.3.1.2. A comparison between the participants' productions in these two phases asserts whether and to what extent the stimuli affected the participants' productions. Sometimes, a third, post-shadowing phase is added to examine whether the effect – or the lack thereof – found in the shadowing phase remains when the external inputs are absent. Figure 5.4 shows a complete flow of a shadowing experiment. It is important to note that the baseline for change is *not* 50 % (randomly retaining preferred realization or adopting stimulus' form). Since people are not likely to spontaneously change their speech style without a reason, the assumption here is that the frequency of such changes represents the external influence of the stimulus on the speaker. More generally, the degree of accommodation can be seen as a speaker's tendency (or, more formally, the *probability*) to converge to an interlocutor.

The ability to present specific contrasts and measure production differences in a supervised fashion makes shadowing tasks suitable for study accommodation and they are often used in vocal accommodation experiments, in which participants are asked to re-produce utterances uttered in stimuli or by a human interlocutor (e.g., Shockley et al., 2004; Babel et al., 2014; Walker and Campbell-Kibler, 2015; Dias and Rosenblum, 2016; Pardo et al., 2018). However, this method also has some disadvantages. One drawback is that while it is suitable for a controlled experimental environment, it only loosely represents a real-world conversation, due to the lack of utterance unpredictability, turn-taking mechanism, subjective common goal, and more. Another potential hindrance is that participants might tend, intentionally or not, to imitate the stimuli, which may lead to a false effect of high convergence that does not represent the participants' natural behavior. This should be addressed in the experimental design and the instructions are

given to the participants, by not using wordings like “repeat”, “mimic”, or “like you heard” and by not making the target features too obvious to the participants.

A distinction can be made between two types of shadowing: In *Close shadowing*, participants start their production while a stimulus is still being played, forcing them to deal with both comprehension and generation at the same time. *Consecutive shadowing*, contrarily, requires participants to listen to a stimulus in its entirety before uttering anything themselves. This becomes harder the longer a stimulus is, merely due to the increasing difficulty to remember long segments. Both of the experiments presented in this chapter use consecutive shadowing. The simulated human-computer interaction (HCI) experiment (Section 5.3) uses short sentences with which the participants are already familiar from the baseline phase. In the sung music experiment (Section 5.2) the musical pieces are relatively long, and the analyses accounted also for parts that the participants did not produce (due to memorizing difficulties or otherwise).

5.2 Prosodic alignment in novel and familiar sung music

Although this work deals with vocal accommodation in spoken language, speech is not the only human ability that uses vocal capabilities. Singing has common characteristics with speech, like the production of different sounds (typically forming words), prosodic properties like rhythm and intonation, and others. On the other hand, singing is usually not used for information exchange purposes. These similarities and dissimilarities make singing an interesting subject to vocal accommodation study. Specifically, it can be investigated whether humans show accommodation in certain vocal properties only when used in speech or as a more general tendency related to their use of voice. Since both speaking and singing are used in social contexts, external factors may potentially affect them both. This enables, among other things, convergence effects between people productions to take place. In the case of music, convergence can be expressed in different aspects than in speech, like more accurately singing, shift in the musical key, tempo changes, etc. Some rhetorical aspects of music and spoken language can be described in musical terms. These two vocal capabilities share some properties in both production and perception. Such common properties include articulation rate, intensity, timbre, and more. Moreover, intonation, pitch, timbre, rhythm, and tempo are all common in descriptions of music, as they are in speech (Molino, 2000; Jackendoff, 2009). Similarly, Day-O’Connell (2013) also shows how some phenomena related to spoken language can

also be described using musical means. Another important aspect is that both have a temporal dimension and evolve over time. However, music consists of defined *absolute* pitch and rhythmic targets, making it easy to compare peoples' productions to some ground truth, as oppose to speech, where specific prosodic values are more subjective and dynamic. This is even more salient when dealing with familiar musical materials, as both the singer's and the listener's expectations are already primed (Meyer, 2008). In speech, on the other hand, the phonetic features of a specific utterance are not expected to match specific absolute values. Since the focus here is on vocal changes, sung music was examined in the study. To prevent influences due to the phonetic properties of specific words, the singing was performed without lyrics (see Section 5.2.1.3).

The main research question of the study presented here is whether convergence occurs in singing as well, and, if so, whether specific parts of the musical pieces are prone to changes. Convergence can be realized on the absolute level, meaning that the participants shift their overall pitch range (the *key*) and tempo to be closer to the recording, or relative to their own singing by making the pitch and temporal intervals between the target notes more precise after listening to the recording. A secondary research question is how the familiarity with the musical material affects reproduction. The expectation here is that the participants' performances of the familiar lullaby will be accurate even before listening to a chapters/shadowing/shadowing-experimentrecorded version of it in terms of deviation from the target intervals, but even more so afterwards. When reproducing an unfamiliar melody, it is not expected that the participants will remember it in its entirety, but rather that they would stick to repeating segments or parts with smaller intervals and simpler rhythms.

5.2.1 Experimental design

5.2.1.1 Target features

Phonetic convergence in speech has been studied with respect to various prosodic features, such as speech rate (Pardo et al., 2012; Schweitzer and Lewandowski, 2013), fundamental frequency (Collins, 1998; Babel and Bulatov, 2012), intonation (Simonet, 2011; D'Imperio et al., 2014), rhythm (Krivokapic, 2013), and more. Corresponding to those, the study presented here deals with the musical prosodic features *tonal deviation* (perceived fundamental frequency (f_0) difference), *rhythmic precision* (with respect to specific rhythmic patterns), and overall tempo and key choice. The latter two are global

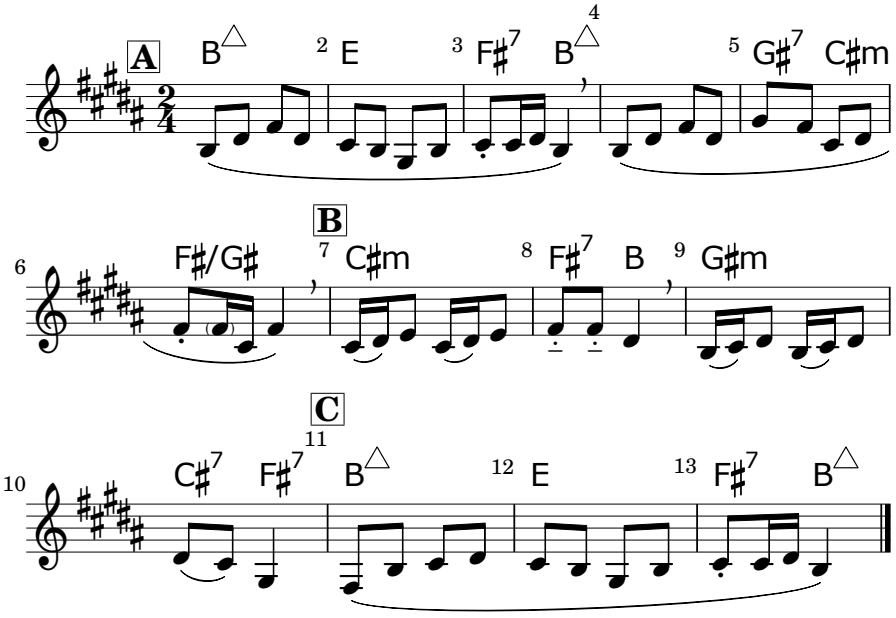
properties and were determined based on an entire performance. In Section 5.2.2 it is explained how these features were measured in music, where the tonal and rhythmic targets are defined based on a musical theoretical framework.

5.2.1.2 Material and participants

Sung lullabies were chosen for this study, as they are more memorable than other musical genres and instrumental pieces, especially among mothers to babies (Trehub and Unyk, 1991; Weiss et al., 2012). When communicating with infants, adults tend to use exaggerated prosody with elevated melodic pitch and distinct rhythmic patterns (Fernald, 1991). This increased use of singing as well as its function as a means of communication with their babies (see Papoušek et al., 1991; Street et al., 2003) made mothers of small babies suitable for this study. Six participants took part in the study, all of which are mothers to recently born babies and with no hearing impairments. For three of them, this was the first child. Their age ranged from 29 to 37 years (mean 35.5 ±3.25) and the age of their babies ranged from one to seven months (mean 4.5 ±3.5). To further homogenize the participants' characteristics, their musical education and experience were controlled as well. None of them had any professional-level musical background and four disclosed they have been singing or playing an instrument recreationally. Since singing is a skill that can be methodically improved, it was required to find participants who don't sing professionally, but still sing in a social context without the direct goal of improving their singing quality. To that end, only mothers who reported that they regularly sing to their new-born babies were selected, because in the pre-verbal phase, parents often sing to their babies. Furthermore, the particiipants' familiarity with the presented known musical piece, a children lullaby (see below), was verified. All the participants reported that they know the lullaby well enough to spontaneously sing it from memory.

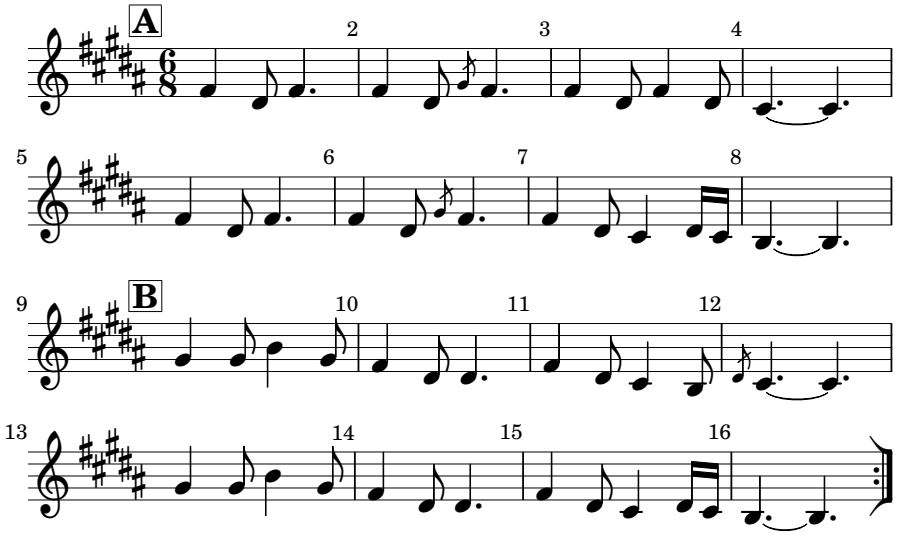
Two lullabies were used, one for each experiment (see Section 5.2.1.3): The first is "Tune for the Yakinton"[14] (hereafter "Yakinton", see Snippet 1), which is a famous Israeli children lullaby. The second is a culturally universal lullaby composed for experimental purposes (Twig, 2016, pp. 22-47, see Snippet 2), which contains cross-cultural characteristics, like repetitiveness, simple melody, and a limited inventory of tonal and

[14]Pizmon LaYakinton, written by Leah Goldberg in 1940; *Yakinton* is the Hebrew name of the Hyacinth plant.



Snippet 1: The Yakinton lullaby transposed to B major. The square labels "A", "B", and "C" mark the *theme*, *bridge* (or *development*), and *recapitulation* sections of the lullaby, respectively. The breath marks are placed where the participants are expected to make a brief break and/or lengthen the ending of a phrase. The first sixteenth note in bar six is in brackets since it is not present in the original melody and was therefore also excluded in the recorded version played to the participants. However, it is common to add it, and indeed all participants included it in both performances.

rhythmic patterns (Unyk et al., 1992; Trehub et al., 1993). Therefore, while the first one was expected to be known to the participants, they could not be familiar with the second one. Both lullabies are short (13 bars at ♩ = 61 (≈26.5 s) and 16 bars at ♩. = 33 (≈58 s), respectively) and in major keys. The lullabies were recorded a cappella by a trained female singer in the same age group as the participants in a professional recording studio at 44.1 kHz sampling rate and 16-bit resolution. To avoid changes in voice production, decrease vibrato, and reduce the singing effort, they were both transposed and recorded in B major, which is relatively low for female voices. This also prevents influences originating from the use of a different key for each lullaby. The syllable [na] was used throughout the lullabies in both recorded versions instead of all lyrics to eliminate biases stemming from the meaning of the words or the realizations of specific sounds. This way, the possibility that participants would hesitate in their performances because they know the melody but not the lyrics was avoided as well.



Snippet 2: The universal lullaby transposed to B major. The square labels "A" and "B" mark the structural parts. The grace notes in bars 2, 6, and 12 were included in the recording but due to their secondary melodic role did not penalize performances that lacked them.

5.2.1.3 Procedure

This study consisted of two shadowing experiments (see Section 5.1 and Raveh et al., 2020). The first experiment examined convergence effects between two performances of the participants: The participants were first asked to sing the familiar Yakinton's melody with the syllable [na] instead of its lyrics (regardless of whether the participant could, de facto, recall the lyrics). Besides that, no specific instructions were given, e.g., regarding the tempo, the key, or any other musical preference. Subsequently, the participants listened to the pre-recorded version of the lullaby via wired over-ear headphones. Following that, they sang the lullaby once more and answered some questions regarding the recorded version of the lullaby, to determine how much it differs from the one in their mental memory. Importantly, no reference to either their previous production or the recorded version was made by using wordings like "repeat", "mimic", "like before", etc. The second experiment comprised only a shadowing performance, as the participants were intentionally unfamiliar with the universal lullaby and therefore couldn't produce it without hearing it first. This experiment tested which prosodic features would be replicated more accurately. After listening to the pre-recorded version of the lullaby, they were instructed to sing it themselves to the best of their ability. This required not

only their singing capabilities, but also their musical memory. Admittedly, it was not likely that the participants would remember all parts and facets of the musical material of this experiment. As explained in Section 5.2.1.2, this lullaby was composed using universal characteristics of the genre and should therefore contain similar melodic and harmonic contents to the lullaby in the first experiment. The two experiments were carried out consecutively. In addition to the short questions in the first experiment, the participants also answered a personal questionnaire before starting the second experiment and a closing questionnaire at the end. The entire procedure lasted 15 min to 20 min per participant.

5.2.2 Analyses and results

Since the participants aimed to produce specific musical notes (as opposed to non-specific absolute frequencies in speech production), tones were used for measuring pitch instead of raw Hertz values. For that, quarter tones (QTs) were used instead of semitones to increase the tonal resolution. Using QTs rather than traditional half-tones enables a more fine-grained analysis that can capture more subtle tonal deviations to better analyze the participants' performances. The segmentation of the performances into individual tones was done manually by a trained musician. Silences, non-singing, breaths between phrases, etc. were segmented as well. The tones were determined by the median of the measured frequencies during the tones' duration, excluding the first and last 10 % of the tone segments. This excludes transitions between tones and smooths out vibrato and ad lib ornaments. These values were extracted using Praat (Boersma, 2018) with manual corrections where necessary. Subsequently, the note assigned to each singing segment was determined by selecting the closest QT to the measured frequency in the corresponding segment. This stands in line with the assumption that people sing with a specific tone in mind rather than a frequency. The mapping between tone frequencies and QTs was done relative to the middle A tone, using the formula (adapted from De Klerk, 1979)

$$frequency(QT_n) = 440 \cdot \sqrt[24]{2}^{n}, \tag{5.2.1}$$

where n is the number of QTs away from the middle A tone and 440 Hz is the frequency of middle A based on the equal temperament. QTs are denoted here with the symbols 𝄲 and 𝄳 for one QT and three QTs above a note, respectively. Ultimately,

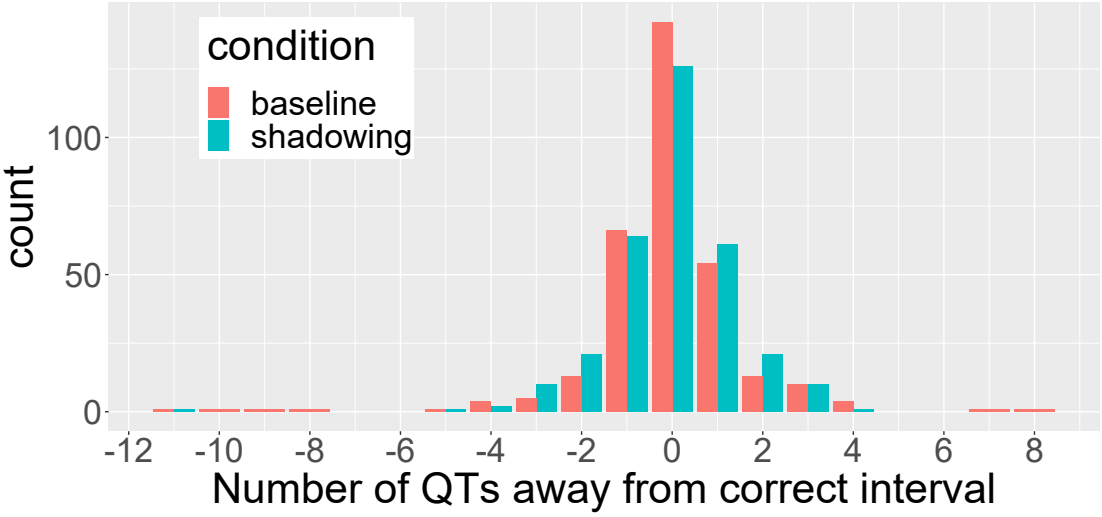


Figure 5.1: A comparison between each participant's distribution of deviations in both phases. The width of the violin shape indicates the counts of the QTs deviations. The upper and lower hinges of the white boxes within the violin shapes show the first and third quartiles, with the thick black line marking the median value. The vertical line crossing the white box show the value range, excluding outliers.

tonal deviations were measured per interval, rather than per tone, as the latter would depend on the key the participants chose, while the former measures tonal accuracy independently of a key. Tempo was measured for an entire performance, taking into account only singing segments. This ensures that pauses between phrases do not influence the perceived singing tempo and that occasional, non-written lengthenings like short ritardandi or fermate at the end of phrases do not mark a specific note as being out of rhythm. Tempo was measured in beats per minute (BPM), which is directly derived from the standard musical notation ♩ =, using the formula

$$BPM = \frac{N + \delta}{overall\ duration} \cdot 60, \tag{5.2.2}$$

where N is the number of beats in the lullaby (26 in Yakinton; 32 in the universal lullaby) and δ is the number of beats added by a participant. Such additions occurred in the participants' performances exclusively, if at all, at the end of phrases (bars 3, 6, 10, and 13 in Snippet 1), but are not present in the pre-recorded version.

As expected, the participants could, for the most part, accurately produce the Yak-

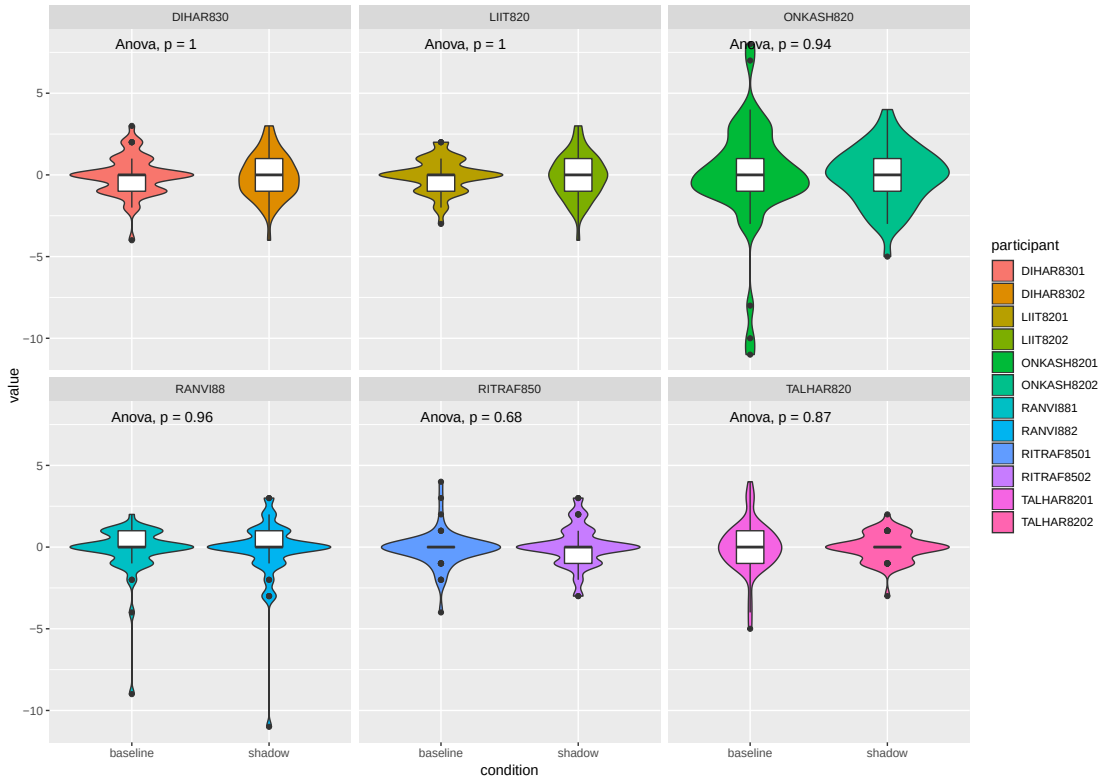


Figure 5.2: Comparison between the distribution of deviations from the correct intervals in baseline (red) and shadowing (blue) performances. The numbers on the x-axis are the number of QTs above or below the correct interval.

inton lullaby in the baseline phase based solely on their memory. However, as Figure 5.1 shows, these performances included several large deviations of two tones or more, which are not likely to be caused by coincidental imprecise singing. In the shadowing phase, in comparison, there was only one such large deviation in all the performances. This adjustment of obviously wrong tones was presumably driven by the exposure to a correctly sung version. Other than these corrections, the deviation distributions shown in Figure 5.1 are roughly symmetric and similar in both phases. Surprisingly, the baseline performances had more correct tones as a whole. It seems, therefore, that the reference version helped the participants to sing within a more accurate range of tones, but somewhat eroded their precision in some notes. To confirm that the changes were subtle and were ascribed mostly to larger deviations, a distributional comparison between the baseline and shadowing productions for each participant. These comparisons are shown in Figure 5.2. The very high statistical distribution similarity results tests confirm that all

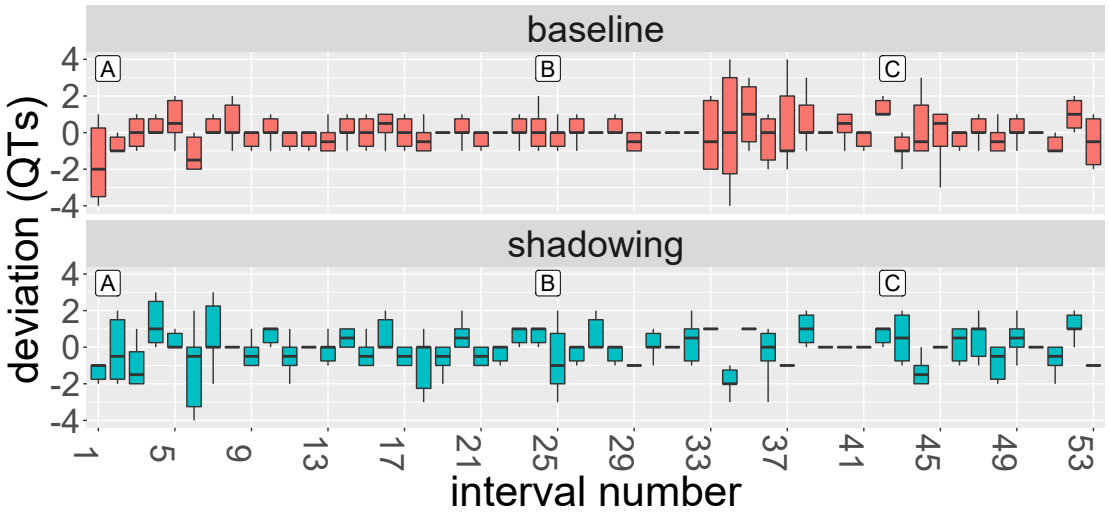
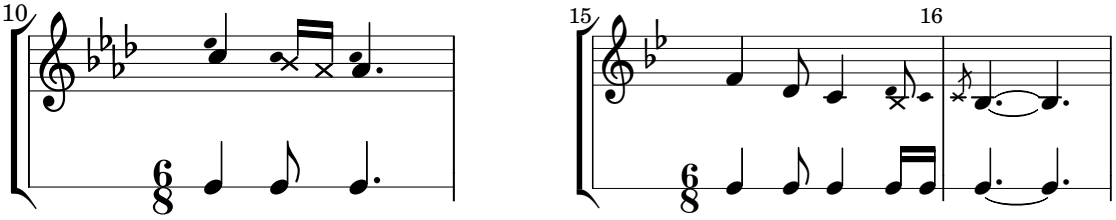


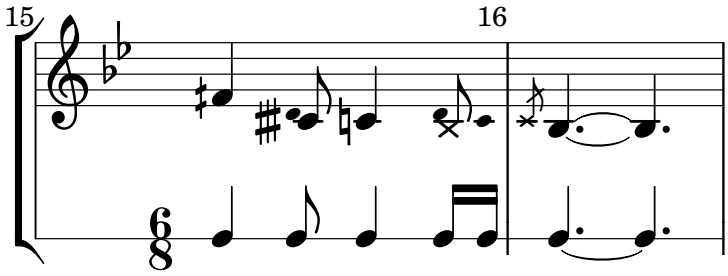
Figure 5.3: Comparison between the deviation distribution of each interval in baseline (top) and shadowing (bottom) conditions. The numbers on the x-axis are the interval indices representing the 53 intervals in the Yakinton lullaby. The distances between the intervals sang by the participants and the correct intervals are shown on the y-axis (outliers are omitted). The labels "A" to "C" mark the different parts of the lullaby (and correspond to the same labels in Snippet 1).

participants didn't substantially change their singing, but, in most cases, the deviation distribution in the shadowing phase was less scattered. The tone-by-tone comparison presented in Figure 5.3 sheds more light on these differences. It is evident that except for the very first interval, the participants showed greater consecutive variation in the second part of the bridge (label "B" in Snippet 1, notes 34 to 42), while in the shadowing condition the first phrase (first seven intervals) showed a similar tendency. Although the bridge moves to a new tonal center, it is not clear why only its second part would cause the singers to be less precise. As for the higher variation at the beginning of the shadowing performances, this might point to the process of re-finding the right tones in the participants' key of preference. This explanation is supported by the key comparisons in Table 5.1, which show that there was virtually no key change between the baseline and shadowing performances for any of the participants. Despite that, the unstable beginning of the shadowing performances indicates that listening to the recorded version influenced the participants' tonal accuracy. This stands in line with the claim that the speech at the beginning of a conversation is most prone to inter-speaker influences (e.g. Orlob, 2018). Participants needed about one whole phrase to overcome this influence



Snippet 3: Examples of tonal (top staves) and rhythmic (bottom staves) deviations in bar 10 (left score) and bars 15-16 (right score) of the universal lullaby. Smaller, stemless notes mark the correct notes where deviation occurred. Crossed-head notes mark those that deviate from the correct rhythmic pattern.

and enter their preferred tonal center anew. Interestingly, the only participant who sang in the same key as the recording did change key in the second performance. The accuracy of tonal replication was measured in two ways, viz. directionality and quantity. First, the correctness of the contour direction in each interval was evaluated (higher tone, lower tone, or same tone). Second, the size of each interval was compared with the correct interval. The participants correctly produced the contour direction in 70 % of the intervals they replicated. The intervals themselves, however, were correct only in 44 % of the time. This shows that the overall contours of the lullaby are more easily recalled than the specific intervals, as would be expected. Snippet 3 shows concrete examples of tonal and rhythmic deviations and Snippet 4 shows the average deviations of all productions. The tempo of the recorded version is 61 BPM. In their baseline performances, three participants sang faster than that and three more slowly. All participants changed their tempo so that it was closer to the recorded version (see Table 5.1). Moreover, the absolute distance from the recording's tempo decreased in all cases but one, which indi-



Snippet 4: Average deviations in the participants' performances in the universal lullaby. Smaller, stemless notes mark the correct notes where deviation occurred. Crossed-head notes mark those that deviate from the correct rhythmic pattern.

Table 5.1: Comparison between the singing tempo and key in baseline and shadowing performances of each participant. The values on the left and right under the key and BPM columns are for baseline and shadowing performances, respectively. BPMΔ shows the BPM difference between baseline and shadowing, with the value in parentheses standing for the change in the difference from the recording's tempo. A negative value means that the participant decreased the distance to the recording.

Participant	**Key**	**BPM**	**BPMΔ**
RITRAF85	F | F♯	76 | 70	6 (−6)
TALHAR82	B | B♭	57 | 63	6 (−2)
RANVI88	A | A	59 | 63	4 (0)
ONKASH82	F♯ | F♯	76 | 69	7 (−7)
LIIT82	F♯ | F♯	59 | 66	7 (+3)
DIHAR83	F♯ | F♯	62 | 61	1 (−1)
recording	(B) | B	(61) | 61	

cates a clear alignment effect. In contrast to the first lullaby, it was not expected that participants would be able to completely replicate all rhythmic patterns in the second lullaby. Two participants replicated part A, part B was replicated by three participants, and one participant managed to replicated both parts. The replication rate of each rhythmic pattern was measured separately. Table 5.2 summarizes the occurrences of the rhythmic patterns (R1–R4, corresponding to the rhythmic patterns in bars 5, 9, 15, and 16 in Snippet 2, respectively) in the original and replicated versions. The proportion of each pattern within a part was generally preserved in the participants' performances, with the expected occasional confusions between R2 and R3 in part A due to their difference only in the last third beat that may be interpreted as a stylistic choice. It is also evident that R1 and R4 were replicated more accurately. An explanation for that is their simplicity compared to R2 and R3, the smaller number of intervals compared to R2 and R3, and that they appear at the beginning and end of every phrase, potentially making them easier to remember.

Table 5.2: Comparison between the percentage of occurrences of each rhythmic pattern in the original and replicated versions in all bar-level patterns. Parts A and B refer to the labels with the same letters in Snippet 2. Each replication row refers to the average over all participants who replicated that part.

	R1	**R2**	**R3**	**R4**
original part A	50	12.5	12.5	25
replications A	54	18	7	21
original part B	25	37.5	12.5	25
replications B	25	42	8	25

5.3 Segmental convergence to natural and synthetic stimuli

After finding accommodation effects in human-human interaction (HHI) in Chapter 4) and other non-speech vocal productions in Section 5.2, the next step on the way to accommodation in human-computer interaction (HCI) is an experiment that tests whether similar effects can be found when humans interact with synthetic voices. The motivation for such an experiment originates from the Computers Are Social Actors (CASA) paradigm (Nass et al., 1994; Nass and Moon, 2000, and cf. Section 2.3.1). If computer-based interlocutors are perceived as social entities in communicative interactions, the question arises whether social-oriented effects, such as accommodation, would occur in HCI as well. An altered version of the experiment presented here is replicated in Section 10.4 as a demonstration of an accommodative spoken dialogue system (SDS).

5.3.1 Experimental design

5.3.1.1 Target features

Three phonetic features of the German language were examined in the study, as listed below. Their variations have been defined as a two-way categorical distinction, corresponding to their perception by humans, even if two of them varies on a gradual scale. Although these features may pass as light dialectical markers (Mitterer and Müsseler, 2013), they do not carry any difference in meaning, and are generally ascribed to personal preference and speaking style.

[ç] vs. [k] at a word-final ⟨*-ig*⟩ syllable

These variations of the phoneme [ç] are both common native speakers of German. Using one variation or the other does not change the meaning of the word. Although [ç] is generally more commonly used in the south of Germany and [k] in the north, they do not mark a specific dialect or socio-economic status. This "neutrality" makes this feature a good candidate, since the experiment does not aim for changes in pronunciation liked to one dialect or the other or an attempt to match a certain social status. It is noteworthy that although the [ç] variation is considered to be the standard, both variations are accepted and people typically do not notice which variation they and their interlocutors use. This feature is treated here as bi-categorical in nature. The very few instances of other fricatives, such as [ʃ] and [j], were counted as [ç] as well, making the distinction practically between fricative and plosive realizations. Here are two examples of sentences with this feature that were used as material for the experiment's stimuli (see Appendix A for the full list of stimuli):

1a)	Der	köni**g**	hält	eine	Rede.
	The	*king*	*held*	*a*	*speech.*

2b)	Ich	bin	süchti**g**	nach	Schokolade.
	I	*am*	*addicted*	*to*	*chocolate.*

[eː] vs. [ɛː] realization of the mid-word grapheme ⟨*ä*⟩

These two phonemes represent the two perceived realizations of this feature's. Vowel quality, as opposed to the [ç] vs. [k] feature, it is not categorical, but gradual. That means that the actual realization can be anywhere between these two realizations. However, despite the gradual nature of vowel quality, native speakers still perceive this feature as categorical (either [eː] or [ɛː], cf. Kuhl, 1991, 2004). This feature is treated here as categorical in production (see Section 5.3.1.2), but as gradual for the analysis purposes (see Section 5.3.2). This allows the detection of bot within-category and cross-categorical changes between productions, which are important for characterizing the convergence process. The [ɛː] variation is in general more typical for the southern federal states of Germany, while [eː] is more common in the north. As in the case of the [ç] vs. [k] feature, the use of one realization or the other (or any in-between them) does not make

any difference in meaning. Here are two examples of sentences with this feature that were used as material for the experiment's stimuli:

1a)	War	das	Ger**ä**t	sehr	teuer?	
	was	*the*	*device*	*very*	*expensive*?	
2b)	Ich	mag	die	Qualit**ä**t	deiner	Tasche.
	I	*like*	*the*	*quality*	*of your*	*bag.*

[ən] vs. [n̩] at a word-final ⟨*-en*⟩ syllable

Unlike the two previous features, this feature does not typically show variation, especially in spontaneous speech. The [n̩] variation is by a large margin the dominant one. The [ən] variation may occur when a speaker wants to emphasize a word/syllable or speak especially clearly, e.g., in a noisy environment. It is rare to hear consistent productions of a [ən] in an ending-syllable ⟨-en⟩. This is true across-dialects and regions, and it is ascribed to the phonological rule *schwa elision* that occurs in the German language, as follows (adapted from Benware, 1986, pp. 142–143):

$$\text{ən} \longrightarrow \varnothing\text{n̩} \,/\, +\text{consonantal} \_\_\_ \ \#. \tag{5.3.1}$$

Here are two examples of sentences with this feature that were used as material for the experiment's stimuli:

1a)	Wir	besuch**en**	euch	bald	wieder.	
	We	*will visit*	*you*	*soon*	*again.*	
2b)	Sind	die	Küch**en**	immer	so	groß?
	Are	*the*	*kitchens*	*always*	*so*	*big*?

It is important to note, that although speakers may have their preferred variants in the contexts given in this study, [ɛː], [eː], [ç], [ɪk], [n̩], and [ən] are all part of the phonetic inventory of native speakers of German and are used by all speakers in other contexts.

5.3.1.2 Procedure

The experiment consisted of three production phases (see Figure 5.4): *baseline* production, *shadowing* task, and *post* production. In the baseline phase, the participants were asked to read out the stimuli from a monitor. Each stimulus was presented separately with nothing else on the screen. The participant's most frequent variant of each target feature (Section 5.3.1.1) was recorded. No instructions whatsoever were given regarding the pronunciation of the sentences. Then, in the shadowing task, the participants produced the stimuli sequentially, each after listening to another voice (either natural or synthetic, both male and female, see Section 5.3.1.3) that used the opposite category of the relevant target feature. For example, if a participant mostly produced [ç], the stimuli's realization of the [ç] vs. [k] contrast was [k]. Based on the production in this phase, the participant's tendency, pace, and degree of convergence were analyzed. These analyses are the basis for the model presented in Chapter 7. Finally, in the post phase, the participant once again read out the stimuli from a screen. The purpose of this phase was to examine whether a convergence effect was maintained when the external input was absent. Between the baseline and shadowing productions, the participants had a break of about seven minutes. Its purpose was to let the mental representation of the production fade, so that the base production will not influence as much on the productions in the following parts. To boost this process, the participants played a game with strong visual aspects and non-verbal sounds only. Conversing with the participant was avoided as much possible as possible in order to prevent other verbal input from influencing their mental representations. The participants' performance in the game was not recorded and did not influence the next parts in any way. A program developed specifically for the purpose of this experiment was used for its execution. It included functionalities tailored for this experiment and a graphical user interface (GUI) the experimenter could use during the and between the phases to quickly record participants' performance and prepare the next phase. These include setting up the participants' audio streams (e.g., playback and recording volumes), extracting the appropriate stimuli from the database, semi-randomizing the stimuli into balanced groups, logging the timing and participants' productions, and more. The experiment was carried out in a sound-proof booth located inside a recording studio. Seeing that the experiment dealt with the way people change their way of speaking based on the speech of others, conversation with participants was kept to a minimum before the experiment began and was avoided till its end.

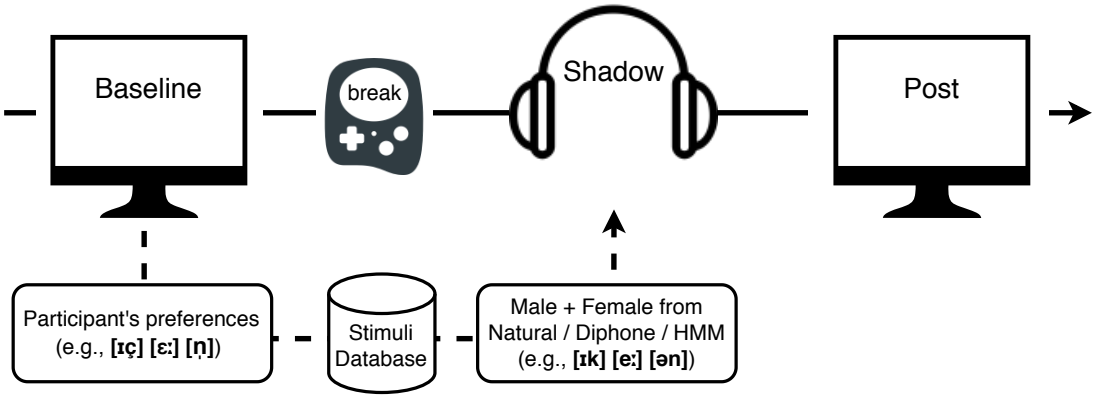


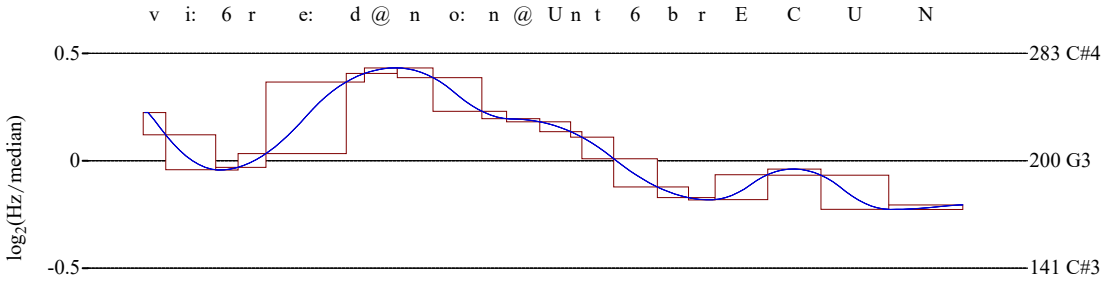
Figure 5.4: Flow of the experiment (from left to right). Stimuli are read from a monitor in baseline and post phases, while heard over headphones in the shadowing task. The participant's preferred variation recorded in the baseline productions are used to select those with the opposite realizations from the stimuli database for the shadowing task. Each participant listened to both male and female voices of one of the stimulus types natural, diphone, or HMM.

5.3.1.3 Stimuli and participants

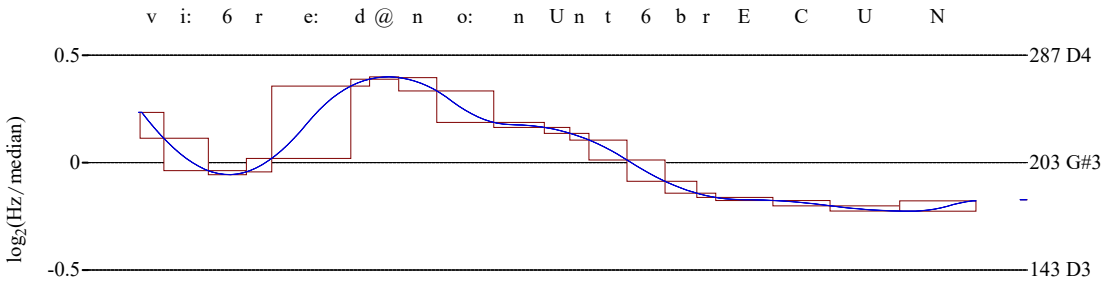
As mentioned above, shadowing experiments are commonly used in accommodation studies. However, this is typically done with single words as targets or with a repeating carrier sentence. Contrarily, in the experiment presented here, participants shadowed full sentences instead of single, typically mono- or bisyllabic (non-)words. Each of the three target features was represented in three declarative and two interrogative grammatical German sentences, five to seven words long. Additionally, 25 filler sentences, in which none of the target features occur, were introduced as a control mechanism and to not make the target features too obvious. At the beginning of the baseline phase, five additional filler sentences were shown to let the participant get used to the task and the setting. Three sets of stimuli were created (one natural and two synthetic), leading to a total of 270 stimuli ((3 features × 5 sentences + 25 fillers + 5 warm-ups) × 3 sets × 2 genders). See Appendix A for the full list of stimuli. One set of stimuli was created with the (natural) speech of a 25 years old female and a 25 years old male native German speakers. As for the synthetic stimuli, there are various synthesis methods to choose from: Formant synthesis (e.g., Burkhardt and Sendlmeier, 2000), unit selection (Hunt and Black, 1996; Black, 2003), diphone synthesis (e.g., Dutoit et al., 1996), and probabilistic (e.g., using hidden Markov models (HMMs) as described in Zen

and Toda, 2005; Zen et al., 2009), to name some. Diphone synthesis using Multi-Band Resynthesis Overlap-Add (MBROLA; Dutoit et al., 1996) and probabilistic HMM synthesis were selected for creating this experiment's stimuli, due to their combination of control over pronunciation and overall quality. One stimulus set including both male and female voices was created with each of these methods. These three sets were stored in a database that was used in the shadowing phase of the experiment (see Figure 5.4). A more advanced technique, like the neural sequence-to-sequence (seq2seq) used in Section 9.2.2, was not selected due to its lack of direct control over specific segments and to avoid voices resembling natural too much natural voices. To better focus on segmental differences, suprasegmental properties in the synthetic stimuli were fixed to match those of the natural utterances. This was done separately for male and female voices with the respective human speakers. The fundamental frequency (f_0) contour (and by extension also stresses) and segment lengths (and by extension also speech rate) of each sentence of the natural stimuli were imposed on the synthetic stimuli in both synthesis methods (and see Raveh et al., 2017a). With MBROLA this process is straightforward, as the duration and pitch values can be directly passed as input parameters. For the HMM synthesis, the process was more complex. To predict voiceness, the mel-generalized cepstra and band aperiodicity coefficients were first extracted from the spectrum of the output signal of the regular HTS process. Subsequently, A neural network with a hidden layer of 128 neurons was used to predict the voicing property from the cepstra coefficients. Then, a voicing mask was applied to the imposed f_0 contour to obtain the final f_0 coefficients. Finally, All the coefficients were used to generate the output signal in a standard synthesis chain with a mel log spectrum approximation filter and the STRAIGHT vocoder (Kawahara, 2006). The segment lengths were directly taken from the annotations, and the f_0 contours were acquired by first interpolating the contour of the natural stimuli and then record the f_0 value at the beginning and the middle of each segment. It goes without saying, however, that due to the limitations of the synthesis techniques, the generated contours were not always *completely* identical to those of the corresponding natural stimuli, as shown in Figure 5.5. Nevertheless, no substantial differences in overall sentence intonation or stress were introduced.

56 native German speakers from ten different states in Germany took part in the experiment. Three non-overlapping subgroups listened to the three stimulus sets. In a post-experiment questionnaire, 80 % of the participants indicated that change the way



(a) f_0 contour of the natural production



(b) synthetic stimulus with imposed f_0 from the natural production

Figure 5.5: MOMEL-INTSINT contours of the natural stimulus (top) and its corresponding MBROLA synthetic stimulus (bottom) for the sentence *"Wir reden ohne Unterbrechung"* with corresponding SAMPA transcriptions. The numeric and alphanumeric values to the right are the absolute pitch frequencies and their corresponding musical tone. The scale to the left displays one octave around the median pitch of the signal.

they speak depending on their interlocutor, 50 % believed they would converge to an interlocutor of the same dialectal background, and 15 % claimed they would converge to an interlocutor from a different dialectal background. Each participant was presented with only one of the three stimulus types. Table 5.3 shows a detailed overview of the participants. Importantly, the ages of the human speakers that recorded the natural stimuli match the mean age of all participants. As explained in Section 5.3.1.2, the participants' preference for each of the target features was obtained in the baseline phase (as summarized in Table 5.4). At the end of the experiment, the participants were asked which realization of each feature they believe to produce themselves and what they think of the other version of it. About 75 % reported a positive attitude towards the version they *do not* believe to produce themselves. Only a minority of participants showed a negative attitude towards the other variation. Based on the results presented in Section 5.3.2, it seems plausible that a positive attitude towards the features entails

Table 5.3: Summary of participant characteristics listening to each stimulus set.

condition	**participants**		**age range**	**mean age**
Natural	17	female	19 to 33	26
	4	male	23 to 34	30
Diphone	14	female	19 to 50	26
	4	male	23 to 34	27
HMM	13	female	18 to 51	28
	4	male	22 to 37	25

Table 5.4: Summary of participants' preferred realization of each target feature based on productions in the baseline phase.

condition	[ɛː] **vs.** [eː]		[ɪç] **vs.** [ɪk]		[n̩] **vs.** [ən]	
Natural	11	10	12	9	21	0
Diphone	14	4	9	9	17	1
HMM	10	7	6	11	16	1

a higher probability of converging to them.

5.3.2 Analyses and results

The occurrences of each feature were analyzed separately. The feature [ɛː] vs. [eː] was measured as a continuum in the F1-F2 formant space. The first and second formants of each target segment were measured at the temporal midpoint in all productions as well as in the stimuli using Praat (Boersma, 2018). These values were used for calculating the Euclidean distance between the participants' and stimuli vowel realizations in each sentence, using the formula

$$E_{dist} = \sqrt{(F1_{participant} - F1_{model})^2 + (F2_{participant} - F2_{model})^2}. \tag{5.3.2}$$

Smaller distances in the shadowing or post phases compared to the baseline phase indicate a convergence effect. Figure 5.6 illustrates the convergence effects as realized by one of the participants. The distances were measured relative to the mean values of all stimuli in the set to which the participant listened. These distances were used to fit linear mixed-effects models with phase as a fixed effect, subject and target word as random

Table 5.5: Mixed effects results for the feature [ɛː] vs. [eː] with three stimuli sets. Each column compares the difference between two phases.

Natural	**base-shadow**	**base-post**	**shadow-post**
intercept	69.94^{***}	32.88	-33.67^{**}
	(14.89)	(17.53)	(11.44)
PREFERENCE	33.79^{*}		
	(14.89)		
observations	210	210	209
Diphone			
intercept	−1.68	−1.49	0.44
	(8.72)	(9.93)	(7.24)
HMM			
intercept	32.64^{*}	−2.12	−31.23
	(13.61)	(12.38)	(15.86)

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

effects (intercepts and slopes), and distance as the dependent variable. Models were fit for both preference groups compared by an ANOVA. The differences between baseline and shadow phases were found to be significant in the natural and HMM groups, and between the baseline and post phases only in the natural group. Table 5.5 summarizes the results for this feature. Figure 5.6 provides additional insights regarding the perception of the vowel quality contrasts in the synthetic stimuli due to their realization in these sets.

The feature [ɪç] vs. [ɪk] was evaluated based on the percentage of same-category realizations of the participants with respect to the stimulus set they listened to. If the percentage increased, e.g., between baseline and shadow productions, the participant converged to the stimuli. Figure 5.8 summarizes the per-phase differences for each set. With all three data sets combined, the number of same-variant productions increased by about 30 % from the baseline phase to the shadowing phase, and decreases again in the post phase, but to a lesser degree. That means that all in all, participants not only converged to the stimuli in the shadowing phase, but the effect lasted to some extent in the post phase as well. The statistical significance between the phases within each stimulus type was tested using a Gaussian linear mixed-effects model. For both the natural and diphone sets, the increases between the baseline and the shadowing phases

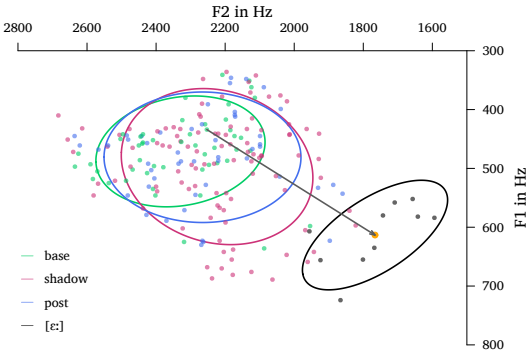


Figure 5.6: An example of a participant's vowel quality convergence towards the stimuli. The participant's preference is [eː] while the variation [ɛː] is used in the stimuli (and cf. Figure 5.7). The colors of the circles represent the **base**, **shadow**, and **post** phases, and the model's formant values with ±1 standard deviation from the bivariate mean. The arrow shows the distance between one of the participant's production and the **model mean**.

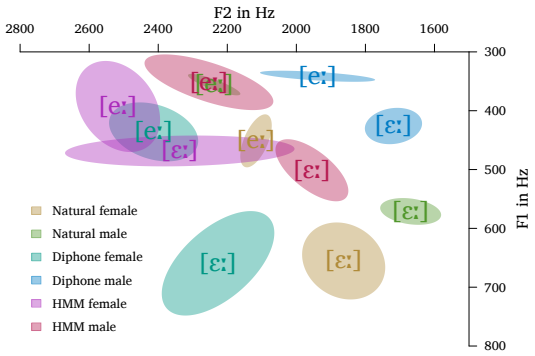


Figure 5.7: Values areas of the first two formants of the three stimulus sets' [eː] and [ɛː] instances. Each color represents both male and female ranges of on stimulus type (natural, diphone, or HMM). The smaller a circle is, the more well-defined the mean target of this stimulus group is. Moreover, the further apart circles of the same color are, the more distinct the difference is within this set.

were significant, 10 % to 39 % and 16 % to 48 %, respectively. Moreover, in these sets, the decreases in the post phase did not go all the way to the baseline level: 39 % to 23 % for the natural set and 48 % to 36 % for the diphone set. For the HMM set, the increase between baseline and shadowing was 8 % to 40 % and did not reach the significance threshold. However, the decrease in the post phase reached the baseline level again and was significant, from 40 % to 12 %.

The feature [n̩] vs. [ən] was measured based on the lengths of the potential schwa segments in the sentences. To decide whether schwa was present or absent in the participants' target word productions, the lengths of relevant segments between the preceding consonant – [d], [t], [ç], [x], or [f] – and the final nasal were measured. A duration of 30 ms was established as the threshold of a perceived schwa segment. This decision is also supported by the fact that all schwas occurrences in the stimuli sets were at least 30 ms long. The segment length range of the natural stimuli, along with the lengths' variance of the participants' productions, is shown in Figure 5.9. Note that only two participants showed a preference toward the [ən] variant in their baseline productions

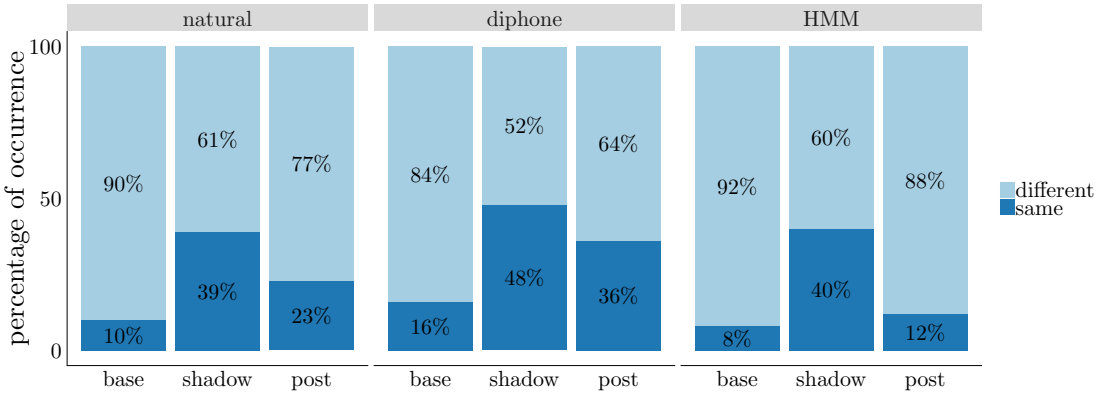


Figure 5.8: Percentage of same-variant (lower numbers, darker background) and different-variant (top numbers, brighter background) realizations of participants with respect to the stimuli they listened to. An increase between the base and the shadow phases indicates convergence towards the stimuli. Similarly, the decrease between the shadow and the post phases back towards the baseline level shows the degree to which the convergence effect remained when the stimuli were not present. The participants listened to stimuli with their dispreferred variation. The same-variant realizations in the baseline phases come from the minority of their realizations, as explained above.

(cf. Table 5.4). This indicates that schwa does not commonly appear in the examined context, which might make it harder to trigger convergence among the participant. Seeing that a statistical analysis for a group of two participants is likely to be misleading, these participants were excluded from the analysis of this feature. Table 5.6 summarizes the percentage of schwa occurrences in the three phases. There was an increase of schwa occurrences between the baseline and shadowing phases for all stimulus types, with the difference being significant for the natural and HMM sets with changes of 9 % and 5 %, respectively. In the post production, the number of schwa occurrences decreased to approximately the baseline level for all conditions. Interestingly, while for the natural stimuli this decrease was still above the baseline percentage, for both the synthetic sets the percentage in the post phase was even lower than in the baseline phase.

5.4 Conclusion

The results found in the music experiment show that alignment occurs in singing, more so with respect to temporal features than to tonal ones. This stands in contrast to findings in interactive speech (e.g., Raveh et al., 2019a). Even so, the results emphasize

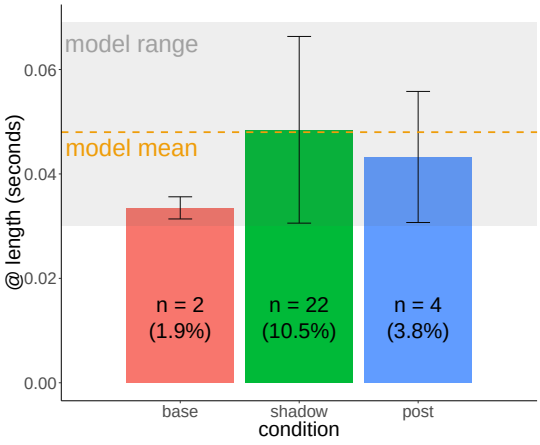


Figure 5.9: Lengths of ə segments in the three phases. The height of each bar represents the average length in this phase, and the corresponding whiskers indicate the overall value range. The gray area shows the value range of the stimuli, with the mean length at the orange dashed line.

Table 5.6: Percentage of schwa occurrences in each phase and stimulus set. N stands for the number of sentences with the target feature [n̩] vs. [ən], which is derived from the number of participant with [n̩] preference (the shadowing phase always has double the number of sentences, because both male and female voices were used).

	base	**shadow**	**post**
Natural	1.9 % $N = 105$	10.9 % $N = 210$	3.8 % $N = 105$
Diphone	2.4 % $N = 85$	4.1 % $N = 170$	1.2 % $N = 85$
HMM	2.5 % $N = 80$	7.5 % $N = 160$	1.25 % $N = 80$

the similarity between the two social oral capabilities. They are therefore also prone to influence each other and can potentially be related and enhance one another. For example, Nardo and Reiterer (2009, p. 216) explain that *tonal* and *rhythmic* abilities are measures of musicality and also related to phonetic talent. This idea is also supported by Tsang et al. (2018), who found a correlation between musical experience and sensitivity to convergence. Similarly, pitch has been found to correlate with the level of agreement between interlocutors in dyadic conversation (Okada et al., 2012). The manner in which distances in vocal behavior decrease or increase may depend on further aspects of the social environment and auditory context, as suggested by Noy (1999). To sum up, convergence to musical stimuli was observed, but not in the same way for pitch and tempo. While tempo became globally closer to the recorded version in absolute terms, the tones were produced more precisely but with no change in tonal range (the key). Additionally, fewer large deviations occurring in the shadowing performances, but the tones in the baseline production were slightly more accurate as a whole. Finally, the simpler, more frequent rhythmic patterns were more correctly replicated by participants. Furthermore, with one exception, participants were not able to replicate the entire lullaby. Interestingly, they remembered either part A or B, but did not mix bars

from both.

As for the HCI experiment, different degrees of convergence were found for each of the three target features. The feature [ɛː] vs. [eː] showed significant convergence effects for the shadowing phase in the natural and HMM stimulus sets. The areas covered by each vowel in Figure 5.7 sheds light on possible reasons for the overall worse performance of the synthetic stimuli. First, the vowels of the same category from the male and female diphone voices occupy a much larger area than those from the male and female natural voices, which results in a less distinct convergence target. And second, all instances that are supposed to be [ɛː] in the female HMM voice are located in the area of [eː]. Hence, in the HMM condition, all participants with preference [eː] actually heard their preferred version of the vowel in half of the trials during the shadowing phase. These lead to the conclusion that the lack of a stronger effect could be due to the acoustic properties of the target vowels in the synthetic stimuli and not necessarily to the synthetic nature of the stimuli itself. The feature [ɪç] vs. [ɪk] was found to be a consistent trigger of phonetic convergence. For all three stimulus types, the participants produced the opposite variant than their preference in roughly a third of the trials. These cases of convergence do not only stem from participants that already showed both target forms in the baseline production, but also from participants that produced only one of the two forms in the baseline phase. This is a strong effect compared to the other two target features. It might be explained by the fact that this feature is categorical by nature, as opposed to the other features which have two defined categories but are realized on a continuum (formant values and segment length). The feature [n̩] vs. [ən] showed a rather small convergence effect. This was expected, as schwa is not usually produced in the word-final sequence *-en*. Nevertheless, in all three conditions more instances of schwa were produced during the shadowing phase than in the baseline and post phase. These productions are mostly attributable to one or two participants per condition were sensitive to this segmental feature nevertheless. It was also observed that apart from the identified group differences, the overall degree of convergence varied considerably among the participants, with some being "resistant" to external influence and others sensitive to them. In conclusion, it can be summarized that humans do indeed converge phonetically when interacting with synthesized speech, even if to a smaller extent. However, the degree of convergence depends on the nature of the target feature. The perceptibility of the target feature in the stimuli is proposed as a possible explanation for the fact

that one of the examined features did not show the same extent of convergence for the synthetic stimuli as for the natural ones.

The two shadowing studies presented here show that vocal accommodation can be found in a controlled experimental environment as part of a quasi-conversational scenario. As a continuation to the findings in Chapter 4, these results demonstrate that vocal accommodation occurs also in non-speech and human-computer settings. These – and especially the latter – prove that accommodation effects are not exclusive to HHIs and thus lay the foundations for further investigation of vocal accommodation in HCI. In chapter Chapter 6, accommodation toward a computer-based interlocutor is examined in conversational, goal-oriented tasks. The great individual differences found in these experiments inspired part of the parameters and approaches used in Chapters 7 to 8. Their importance lies in the ability to define different "characters" (or *profiles*, see Section 3.3.3) based on different human behaviors, rather than letting a system behave the same in every interaction. Furthermore, showing that convergence can occur in segmental features as well emphasizes the importance of control over such properties in synthetic speech for triggering vocal accommodation, which is attributed mostly to HHI. The segmental manipulations demonstrated in Section 9.2.2 show how such differences can be achieved.

Chapter 6

Accommodation in Multiparty Interactions with an Agent

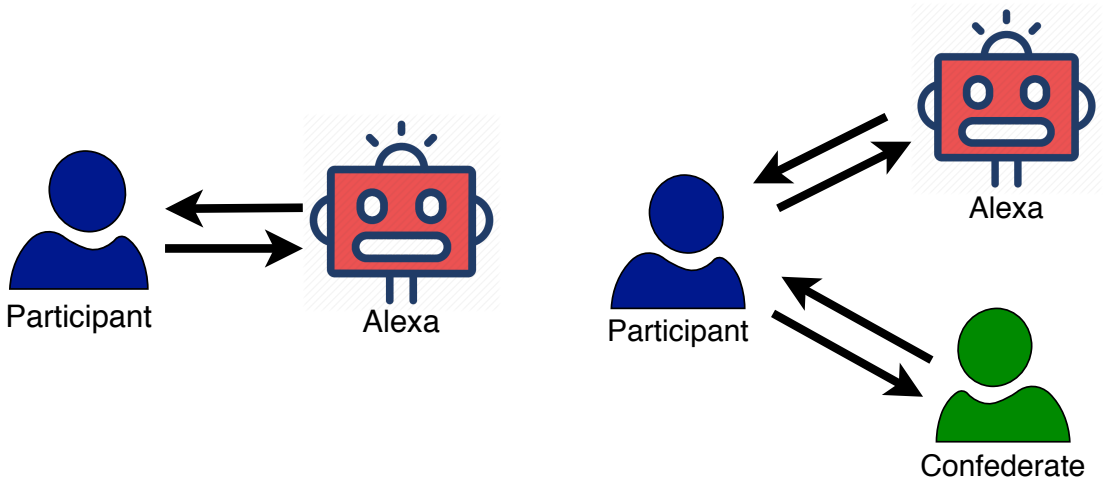
More dynamic vocal behaviors can be established in interactions with multiple interlocutors. This is not only due to the additional possible connections between them, but also because of other factors, like order of speech or the role of each speaker. A human-human-computer interaction study is presented in this chapter, where the effects of different aspects and conditions on vocal accommodation are investigated.

6.1 Speech variations in human-human-computer interaction

Nowadays, we are witnessing an ever-growing presence of devices with spoken interaction capabilities in our everyday lives. As argued in Section 3.2.1, the use of personal assistants (PAs) is rapidly increasing, as more mundane tasks can be achieved using them. The question arises, therefore, whether different speech patterns and characteristics emerge in such human-computer interaction (HCI) compared to human-human interaction (HHI); and if yes, which. The vast majority of experimental work done in the field of vocal accommodation deals with the smallest social interactions, namely dyadic conversations. Those can be dyads of two human speakers in HHI, or a human and a computer-based agent in HCI. Vocal accommodation in these types of interactions is explored in Chapters 4 and 5. However, social interactions may also consist of three or more participants. This is true for both HHI and HCI, but also for interactions with mixed human and computer-based interlocutors and specifically multiple humans talking with a single device. The latter can occur in various situations, like a person consulting a voice assistant (VA) regarding availability in a weekly schedule while setting an appointment with a colleague or two friends ordering tickets from a voice-activated machine. Mixed multiparty interactions already take place in various real-world situations lives. Their popularity – and sometimes necessity – increase alongside the rise in use of conversational AI (C-AI) devices, such as VAs, voice-activated cars, hands-free medical assistants, intelligent tutoring systems (ITSs), social robots, and others. It is important, therefore, to understand whether convergence, divergence, and other effects (like those described in Section 2.1.1) may occur not only in HCIs, but in human-human-computer interactions (HHCIs) as well. Various HCI experiments have shown that participants speak differently to computers and change their speech behavior during the interaction (e.g., Branigan et al., 2010). Some works also compared the reaction of participants to different configurations of the computer-based interlocutor (e.g., Levitan et al., 2016). Yet, none of the above has performed a comparison between human-directed and computer-directed speech within a single multiparty interaction. Moreover, only the influence of the system's speech output on the user speech is typically examined in accommodation experiments, but not the influence of another human interlocutor.

Empirical work on multiparty HCI includes experiment where participants interact with different types of agents, like social robots (Foster et al., 2012; Ibrahim et al., 2019) and human avatars in immersive virtual worlds (Traum and Rickel, 2002). Even in dyadic form, spoken interactions are a hard task for computers. All the more so, when more than one other interlocutors are involved. Measuring accommodation becomes more complex with multiple interlocutors involved, as discussed in Rahimi et al. (2019). There are many technical challenges on the way to realistic, real-time interactions with computers, including – but not limited to – center-of-attention detection, active speaker detection, turn taking, understanding private and shared knowledge, and of course correct speech production and understanding. Interactions with machines are challenging for humans, too, since the former do not behave and react the same way (and often speed) as humans. An example of such a social activity that is reasonably easy for humans to learn but still far from being feasible by computers are social games with a large number of participants. These games typically require the players to be deceptive, track and exploit the behaviors of others, and react quickly to ever-changing dynamics between the players. Jonell et al. (2018) describe the challenges of such scenarios and discusses way to cope with them. The type and severity of those problems depend also on the type of the computer-based agent (see Section 3.2). For instance, embodied agents at least have some basic way to convey non-verbal information, whereas voice-only systems like VAs do not. On the one hand, this gives a wider range of expressions to embodied systems, but requires more communication channels to implement and coordinate on the other hand.

This chapter presents a study that examines interaction-level vocal accommodation in HHCI. In this paradigm, two human speakers work collaboratively with an agent to complete tasks while only one of them can talk directly to the agent. More details about the dataset and the tasks are given in Section 6.2. Investigating such a scenario contributes to the understanding of both the role of an agent in interactions with multiple humans and the influence of another human in a HHCI. These two aspects are examined in the two components of the study: The *addressee* component focuses on the differences between the participant's addressed interlocutor within a conversation (see Raveh et al., 2019c), and the *crowd* component spotlights the influence of an additional human interlocutor on the participant's speech toward the agent (see Raveh et al., 2019a). The question tackled by the second component is whether and to what extent speaking to a second human interlocutor in the same interaction as the agent influences the accommo-



(a) Solo condition (b) Confederate condition

Figure 6.1: Illustration of solo and confederate conditions. A black arrow represents direction of speech from speaker to addressee. Note that the confederate (in green) never talks or addressed by Alexa, but only the participant (blue).

dation toward it, i.e., whether users speak differently towards a VA when another human participates in the interaction. to that end, the distributional and temporal analyses performed in Section 6.3 are based on the participants' *speech directions*, i.e., human-directed speech (HDS) and device-directed speech (DDS), as illustrated in Figures 6.2 and 6.3.

6.2 Dataset

The accommodations study presented here uses the Voice Assistant Conversation Corpus (VACC)[15], introduced by Siegert et al. (2018). This corpus is suitable for this study, because it comprises both HCIs and HHCIs with a 2nd generation Amazon Echo Dot device using the default skill set and German female voice of the voice assistant (VA) Alexa. These configurations are referred to here as *solo* and *confederate* conditions, respectively. Figure 6.1 illustrates the interlocutor relations in each condition. Similar corpora were used to study automatic addressee detection (e.g., Turnhout et al., 2005;

[15]http://www.iikt.ovgu.de/iesk/en/Research+Groups/MDS/Research/VACC-p-4624.html

Shriberg et al., 2013). Although the study here does not set addressee detection or classification as a goal, it aims to provide insights and accommodation-related measures that may be useful for such tasks. In the VACC, the male confederate was present in the room only during tasks in the confederate condition and always sat at the same location. To simulate the spatial situation of a multi-party interaction with a VA, the participant and the confederate sat in similar distance from the device that was situated on a table, roughly forming an equilateral triangle (see Figure 2 in Siegert et al., 2018). The participant had led the interactions with both the confederate and Alexa, and the confederate never talked to Alexa directly. Therefore, in the confederate condition, the participant needed to alternately speak with the confederate and Alexa alternately as part of the same interaction, without explicitly signaling to whom each utterance is addressed. Two tasks were performed in each condition: In the calendar task, the participant's goal was to find available time slots for several hypothetical appointments with the confederate. The participant's pre-defined calendar was stored on the device and was accessible only via inquiries to Alexa. In the solo condition, the participants got written information about the confederate's availability, whereas in the confederate condition, the confederate could be asked about it. The goal of the quiz task is to answer trivia questions, like "When was Albert Einstein born?". Since Alexa was not always able to immediately provide a full answer to all the questions, the required information could be gathered incrementally over multiple turns. Here, the participant solved the quiz alone in the solo condition or teamed up with the confederate so that the two could discuss the question-asking strategy in the confederate conditions. The calendar task was designed so that the way to its solution is relatively straightforward: Query the device for possible times till a match is found. This requires interacting mostly – if not only – with the computer-based interlocutor. Indeed, this task typically elicited interactions, in which the participant interacted with the confederate or the device in discretely separate turn blocks in the confederate conditions. DDS blocks were, unsurprisingly, longer, as the confederate was only addressed when additional information about the task was required. The quiz task's flow was more flexible, since the strategy as to which questions to ask Alexa can be determined by the participant, including the amount and frequency of the confederate's intervention in the confederate conditions. Indeed, more dynamic alternations between HDS and DDS were observed in this task. As a whole, the quiz task is less formal than the calendar task.

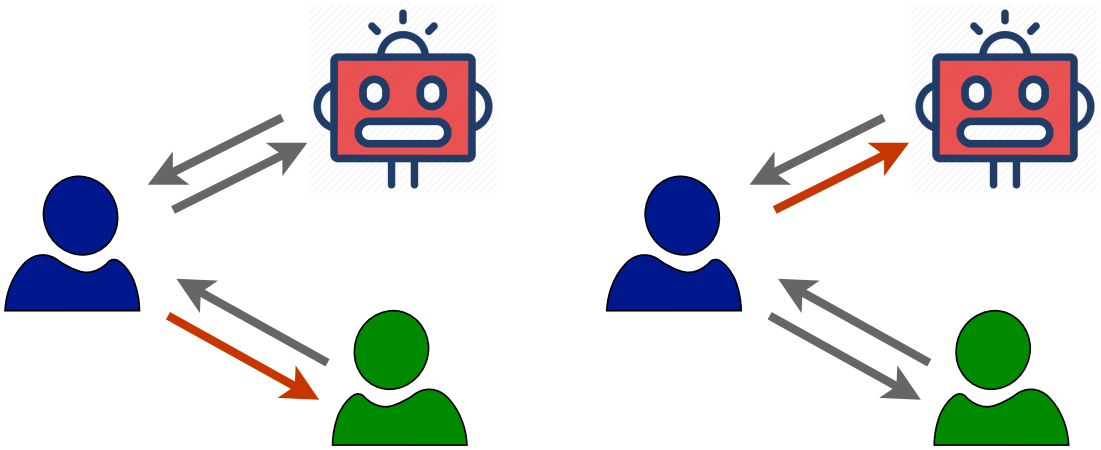
The dataset contains recordings of 27 (14 female) German native speakers in the age range of 20 to 32 years (mean 24 ±3.3). Each participant performed the quiz and calendar tasks in both solo and confederate conditions, for a total of 108 interactions (2 tasks × 2 conditions × 27 participants). These interactions consist of approximately 13,500 utterances, which were manually transcribed and annotated (speaker, speech times, addressee, etc.) and stretch over total recording time of 17 h 7 min (31 min average interaction length). The permutations of the tasks, conditions, and their order were balanced.

Annotations

Each utterance in an interaction was annotated with its speaker, context, and textual transcription. The speaker of each utterance could be the participant, Alexa, or the confederate. Cross-talk was rare, as the participants typically waited till the confederate or Alexa finished talking (except for when they tried to interrupt Alexa mid-utterance if the response was unquestionably wrong or irrelevant due to a recognition error). The context marks the utterance's interaction type, like HDS, DDS, cross-talk, off-talk, laughter, and more. To deal with clearer data, only HDS and DDS contexts were used for analysis, which, together, constituted over 90 % of the interactions' recording time. Transcriptions were obtained using the Google Cloud Speech API automatic speech recognition service and were subsequently manually verified and corrected. Utterances' start and end times were derived directly from the transcriptions' timestamps.

6.3 Analysis

A subset of the VACC was used for the analysis in each component, which is suitable for the speech directions in question. Figures 6.2 and 6.3 illustrate the examined speech directions in each component. Only the 54 interactions from the confederate conditions were used for the addressee component, as the alternations between HDS and DDS within an interaction were examined. For the crowd component, all 108 interactions were taken, because the comparison required executions of the tasks performed in both solo and confederate conditions. Interactions of all 27 participants were included in both subsets. Despite the different sizes of the subsets, the number of comparisons was the same for both components, since the addressee component used both speech directions of



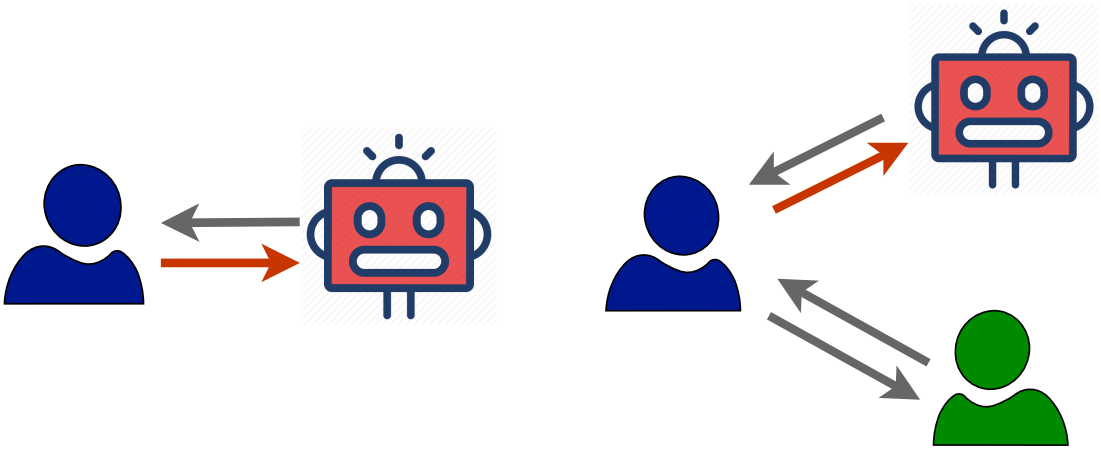
(a) HDS in confederate condition

(b) DDS in confederate condition

Figure 6.2: Illustration of the compared speech directions in the *addressee component*. The orange arrows mark the compared speech directions (participant to confederate and participant to Alexa).

the participants in each interaction ($54 \times 2 = 108$) and the crowd component used only DDS but from both conditions ($54 \times 1 + 54 \times 1 = 108$). These subsets were analyzed based on the audio signals and the annotations described in Section 6.2. The speaker annotations were used to determine to which of the three speakers the measured values should be ascribed. The text transcriptions were only used for verifying the correct audio segments were analyzed. Comparisons were made between the utterances of the same interaction, i.e., within a single task.

To increase temporal resolution, the audio signals were cut into two-seconds *slices*. A single slice always contained audio from a turn of a single speaker. Any remainder shorter than 2 seconds got a separate slice. For example, a turn of length 5.2 s was sliced into three slices of 2 s, 2 s, and 1.2 s. This way, values measured in a slice could belong only to one speaker. Splitting the turns also creates equal, consecutive, and more comparable time units for an interaction without introducing artificial boundaries by dividing it into a pre-defined number of parts (as in Silber-Varod et al., 2018). This is especially important for the temporal analysis (Section 6.4.2). Slice lengths of 0.5, 1, 5, and 10 seconds were experimented with as well. However, those were proved too short to capture changes in articulation rate (AR) (see below), which is dependent on the size of this window, or two long for a comparable and uniform temporal resolution. Two



(a) DDS in solo condition (b) DDS in confederate condition

Figure 6.3: Illustration of the compared speech directions in the *crowd component*. The orange arrows mark the compared speech directions (participant to Alexa with and without the presence of the confederate).

seconds was found to be a good compromise based on these criteria.

The following phonetic features were targeted:

Fundamental frequency (f_0) – mean pitch measured within a slice with hop size of 100 ms and a pre-defined range of 60 Hz to 350 Hz. Gregory et al. (1993) found that this feature is used to produce social similitude and cohesiveness in dyadic interviews. Moreover, this feature showed convergence effects in an auditory naming task (Babel and Bulatov, 2012) and a HCI shadowing task (Bulatov, 2009).

Intensity – mean intensity measured within a slice with hop size of 100 ms. This feature showed significant entrainment effects in game scenarios (Levitan and Hirschberg, 2011) and as a indicator for social desirability (Natale, 1975).

Articulation rate (AR) – the ratio of number of syllables to phonation time within a slice, as described in De Jong and Wempe (2009). Schweitzer and Lewandowski (2013) examined this feature in the context of phonetic convergence, but no effect was found on turn-level and interaction-level analyses.

All features were measured individually in each slice automatically using Praat (Boersma, 2018) scripts. The preprocessing and analysis were performed using the system introduced in Raveh et al. (2018, and see Chapter 10).

Table 6.1: Percentages of interactions in which the distributional difference of each feature was significant

	f_0	intensity	AR
signif. diff.	74 %	89 %	13 %
HDS mean (standard deviation)	10.5 Hz	2.95 dB	0.627
DDS mean (standard deviation)	10 Hz	2.61 dB	0.634

6.4 Results

Two analyses were carried out: distributional and temporal. The first looks at global differences on the interaction level of the participants' productions. The second examines time-based, continuous changes in the similarity between the participants and the other interlocutors.

6.4.1 Distributional analysis

The means, medians, and standard deviations of the target features in the participants' speech in each of the interactions were calculated for both HDS and DDS. These measures shed light on the overall range of values used when the participant was talking to each of the other two interlocutors. They were listed for each target feature chronologically throughout the interaction. These lists were divided into four speech directions based on speaker and context: the participant talking to the confederate, the participant talking to Alexa, the confederate talking to the participant, and Alexa talking to the participant (see four arrows in Figure 6.1(b)). The contrast between HDS and DDS is observable within the participant's speech only, which was active in both contexts. To detect these differences, the distribution of their respective values in the solo and confederate conditions in each interaction pair were compared. This was done by using the two-sample Wilcoxon test (Wilcoxon, 1945), with $\alpha = 0.05$ with the null hypothesis that similar distributions of the target feature were used in both conditions. A significant result of the test means that the participant produced the respective feature differently when interacting with Alexa alone compared to when the confederate participated as well. Table 6.1 shows the percentage of interaction pairs in which the null hypothesis was rejected, i.e., that the feature was utilized differently by the participant in each condition.

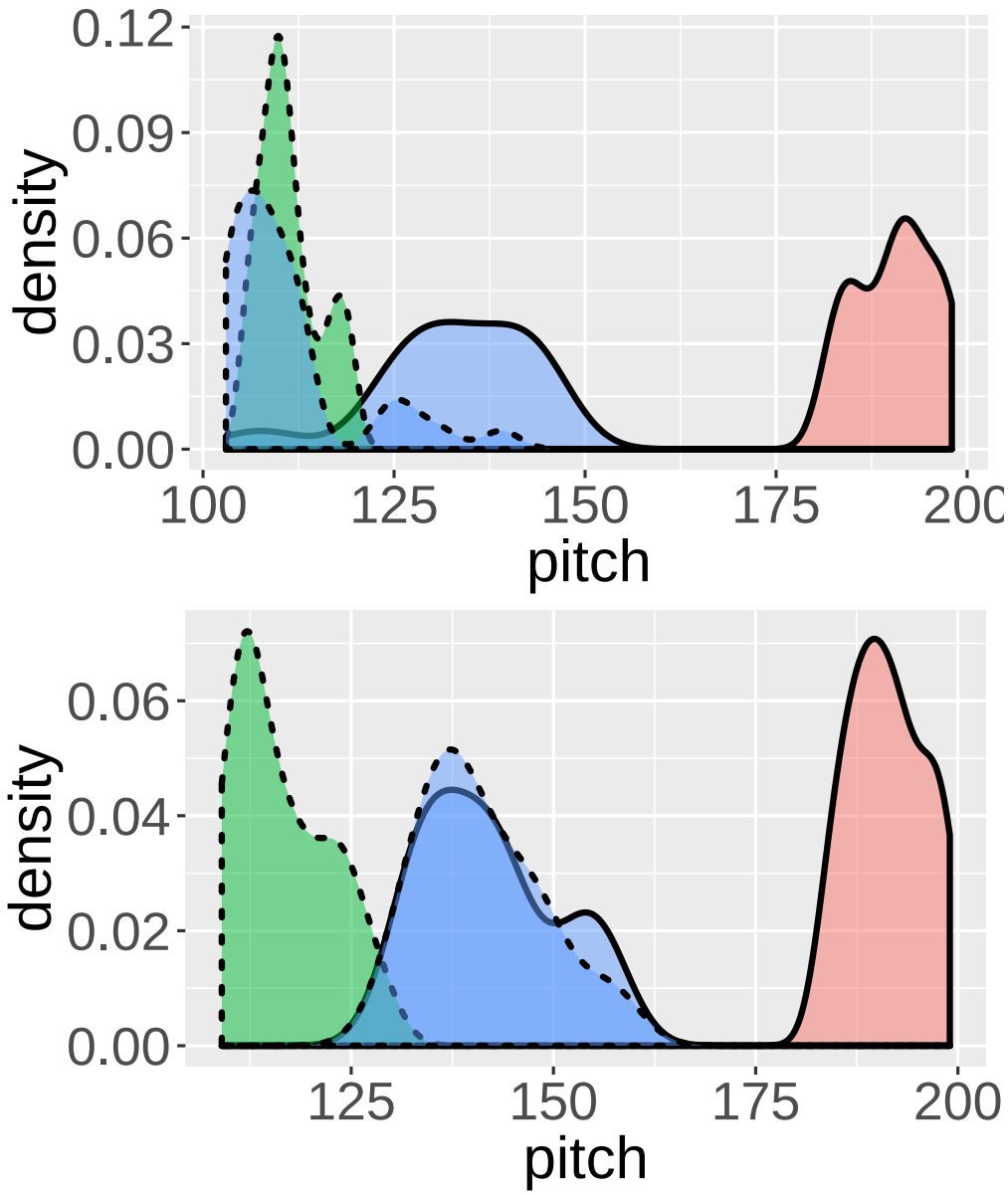


Figure 6.4: Examples of f_0 distributions in HDS and DDS with a *significant* difference (top; quiz task of participant 20171127A; $p \ll 0.0001$, $\alpha = 0.05$) and an *insignificant* difference (bottom; calendar task of participant 20171127C; $p = 0.71$, $\alpha = 0.05$). The colors represent distributions of the participant (blue), Alexa (red), and the confederate (green). The line style differentiates between HDS (dashed line) and DDS (solid line).

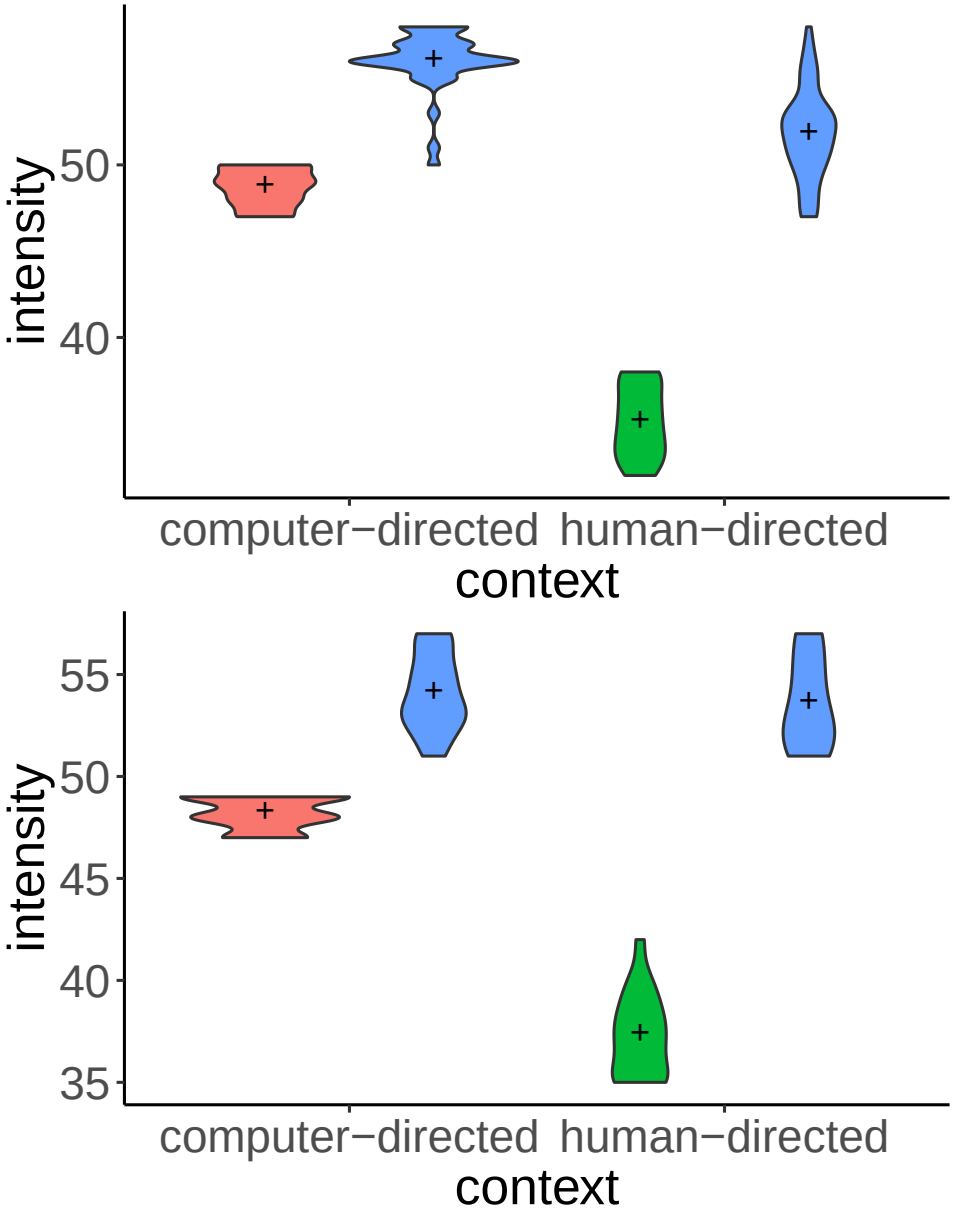


Figure 6.5: Examples of intensity distributions in HDS and DDS with a *significant* difference (top; quiz task of participant 20171127A; $p \ll 0.0001$, $\alpha = 0.05$) and *insignificant* difference (bottom; calendar task of participant 20171127C; $p = 0.55$, $\alpha = 0.05$). The colors represent distributions of the participant (blue), Alexa (red), and the confederate (green). The widths of a boxes represent the value frequencies and the '+' sign marks their respective means.

Table 6.2: Percentage of interaction pairs with significant differences with respect to each target feature with all the interactions together and separated by order tasks.

feature	**any order**	**solo first**	**confederate first**
f_0	67	72	60
intensity	67	76	56
AR	30	31	28

Figure 6.4 show examples of the distributions of a participant's f_0 in HDS and DDS contexts in the addressee component. Since a female voice was always used for Alexa and the confederate was always male, there is a natural gap between their f_0 values. This gap leaves room for convergence to occur, i.e., change in the participants' production in the direction one of the other interlocutors. As Table 6.1 shows, in 74 % of the cases out of the 54 analyzed interactions the difference of the participant's f_0 between HDS and DDS was significant. Out of those, in 85 % of the cases the distribution mass of DDS contained higher values than HDS's, which indicates accommodation towards Alexa. Unlike f_0, absolute intensity values may not be as meaningful due to the device's and the confederate's location relative to the participant's microphone. This means that the absolute values of the participant's intensity in HDS and DDS can be compared directly, but only relatively against Alexa's and the confederate's. Therefore, the differences of this feature as shown in Figure 6.5 should only be compared between the participant's both speech direction within an interaction. These differences between the participants' HDS and DDS were significant in 89 % of the cases. In general, participants tended to speak to Alexa with a louder voice than to the confederate, although their distance from either was the same. The differences between AR distributions in HDS and DDS were significant in 13 % of the cases. This shows that the participants largely spoke with the confederate at the same speed as with Alexa. Interestingly, the AR was temporarily considerably lower when the participants tried to improve intelligibility, especially if the system's output indicated that it did not correctly understand the participant's utterance due to a recognition error. In the confederate component, the performed each task both alone and with the confederate. Since chronologically, by design, one of the conditions needed to precede the other, the percentages were also calculated separately for the cases where tasks were performed first in the solo condition and then in the confederate condition, and vice versa. This separation shows whether interacting first with Alexa

alone, without any human input, influenced the vocal behavior of the participants. As there were no breaks between the components, the only factors for change were the order of the conditions and the involvement of another human speaker. As shown in Table 6.2, the percentages of significant differences when interacting first only with Alexa were indeed higher by 12 %, 20 %, and 3 % for f_0, intensity, and AR, respectively. Figure 6.6 further breaks down the differences between interaction pairs and introduces the factor of the performed task. In line with the tendency shown in Table 6.1, the features f_0 and intensity have the highest percentages of significant cases, regardless of the performed task, and the tasks performed first show higher percentages of different distributions. In the lower percentages, it is the task, rather than the target feature, that shows differences between the cases. And last, for AR, with the lowest percentages, there is a clear difference between the quiz and the calendar tasks. All in all, the `task` factor was a good indicator only for the feature with the lowest difference percentage and the `order` factor was more informative for the features with higher percentages. To sum up, these results show significant differences in the majority of the cases for two of the three features for both the addressee and crowd components. These outcomes provide a look into further aspects of HDS and DDS and speech-related features in HHCI, which may help studies in topics like addressee detection or vocal accommodation in multiparty interactions.

6.4.2 Temporal analysis

Looking at the distributional differences of the target features in HDS and DDS sheds light on the general speech behaviors of the participants. However, this analysis leaves out an important aspect of spoken interactions, namely the time dimension. While the interaction-level distribution measures show accommodation effects based on te overall range and frequency of a feature's values, the temporal analysis adds the information as to how they changed over time. Adding the time dimension gives an overview of an interaction's structure and reveals additional insights regarding its dynamics, such as turn lengths, turn switching, pauses, and accommodation effects. For that, trend lines for the temporal changes needs to be calculated. This was achieved by smoothing the measured value using locally estimated scatterplot smoothing (LOESS) (Cleveland and Devlin, 1988), a non-parametric regression method that deterministically fits a function to a localized subset of the data. The fitting was done for each speaker separately over all

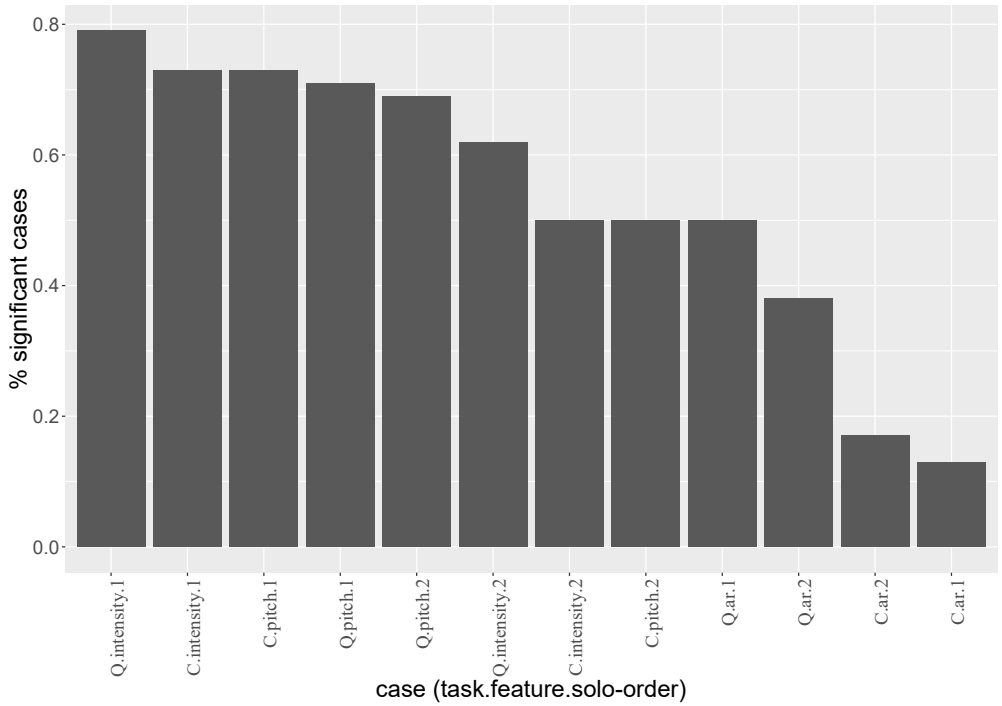


Figure 6.6: Percentages of cases with a significant difference between a feature's distributions in the solo and confederate conditions. A case is a combination of the factors `task`, `feature`, and `order`. For example, the case `Q.intensity.2` contains the comparisons of intensity in interactions of the quiz task where solo condition was performed second.

slices of HDS and DDS with measured values of the features. The first plot in Figure 6.7 shows a case where the absolute f_0 values are roughly the same in HDS and DDS, but the participant's change patterns are different. In the DDS context, the participant generally keeps a stable distance from Alexa's voice, whereas in the HDS context the f_0 values gradually get closer to the confederate's. In both contexts, the participant's f_0 starts around 150 Hz, but in HDS the minimum f_0 is only slightly below this initial value, whereas in DDS it drops as far as 25 Hz lower. A similar example is shown in the second plot in Figure 6.7 for the intensity feature. Unlike the previous example, here the absolute values between HDS and DDS steadily differ by about 5 dB (matching the tendency to talk louder toward Alexa, as described above), but the overall change is similar. That is, in both cases the intensity rises from the beginning to around a quarter of the interaction's duration, and then decreases again until the end, in HDS more quickly than in DDS, down to approximately the same value as at the beginning. Since

Figure 6.7 shows two examples of the quiz task performed by two different participants, it is possible to compare the structure of these interactions as well. As described in Section 6.2, the quiz task in the confederate condition is designed so that the two human speakers need to find an efficient way to solve the questions using Alexa. After improving their strategy, the lead should ultimately be taken by the participant, who interacts with Alexa to solve the questions as quickly and correctly as possible. In both examples, the first half of the interaction contains relatively short turns and rapid addressee changes. This might be ascribed to the fact that the participants are still trying to figure out the best way to interact with Alexa and the confederate. Then, sometime after the middle of the interaction, there is a larger block of DDS, followed by some more turns of HDS where the participants discuss with the confederate about ways to solve the remaining questions. Finally, the interactions end with a shorter block of DDS, in which the participants finish these last questions of the quiz. This structure quiz task was found in most participants' performances.

The accommodation in each conversation were further examined by measuring the contribution of the participant to the overall mutual change. Figure 6.8 illustrated a comparison of a participant's changes in the solo and confederate conditions. The lower part of each plot shows the accommodation changes of the participant during the interaction (blue for convergence and red for divergence), and the upper part shows the floor changes. Note that the confederate condition has fewer floor changes, because the analysis concentrates on the participant and Alexa, and the confederate turns are not shown. The relationship between a feature's values in each slice needs to be determined to describe their temporal changes. To investigate the accommodation effects, a measure for the relative change between slices was used, which calculates the participant's contribution to the overall change in proximity between the participant and Alexa. Alexa's contribution is considered to be a static effect, as it is not deliberately changing its output based on the participant's speech.The degree of change between two slices is calculated by

$$change_t = -\Delta_{t,t-1} \mid S_{part} - S_{Alexa} \mid, \tag{6.4.1}$$

where the index t refers to the current slice and S_{part} and S_{Alexa} are the smoothed values of the participant and Alexa, respectively. The minus sign at the front flips the value so that convergence is represented by positive values and divergence by negative values.

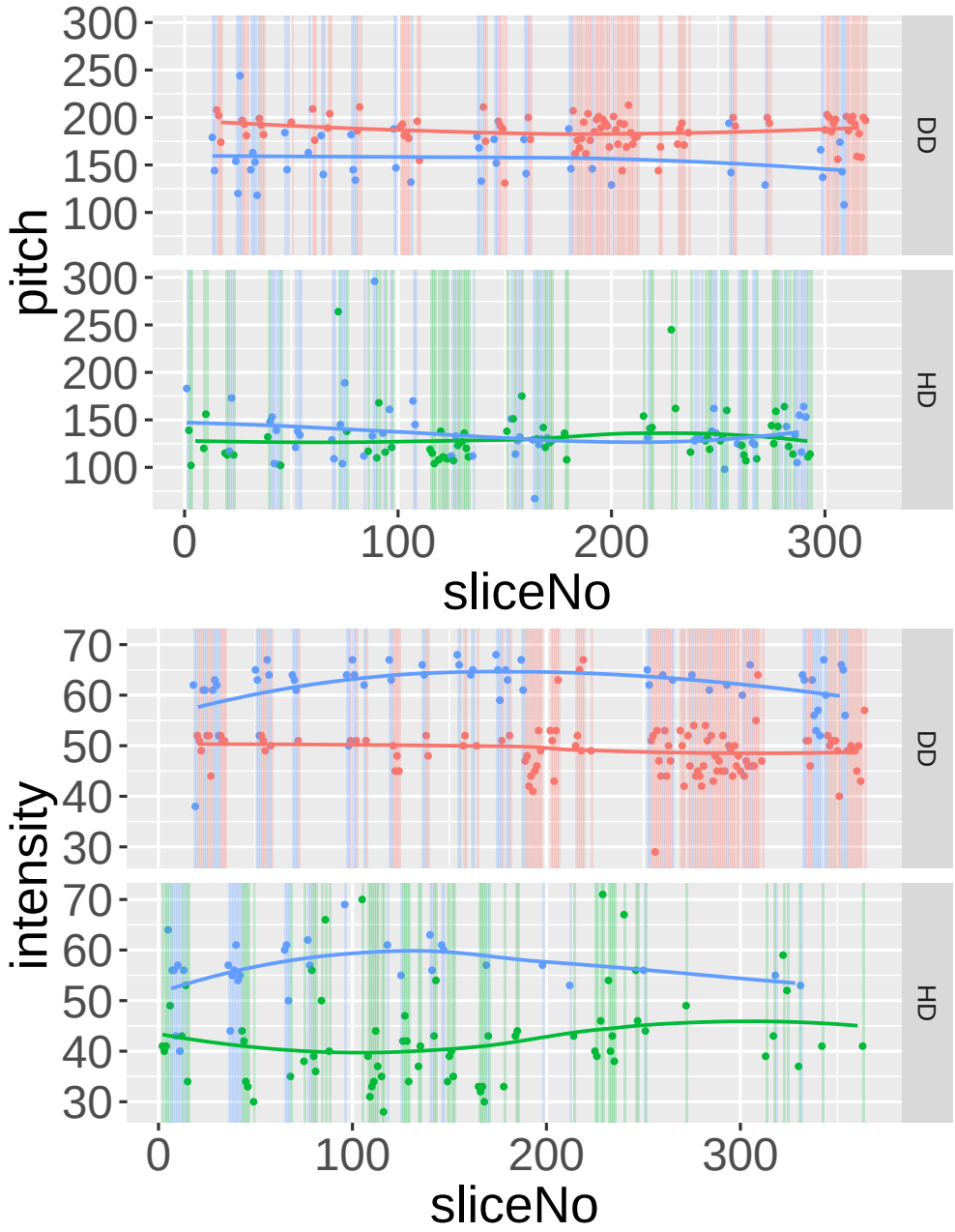


Figure 6.7: The changes of f_0 (top; quiz task of participant 20171129A) and intensity (bottom; quiz task of participant 20171129B) over time in DDS and HDS (upper and lower parts of each plot, respectively). Alexa's voice is marked in red, the confederate in green, and the participant in blue. The timespans on the x-axis are represented by turn slices, as explained in Section 6.3, and the y-axis shows the values of the feature. A slice's background color indicates the speaker in this slice and the dots with the same color show the measured value of the feature in that slice. The trend lines are smoothed values calculated by LOESS (Cleveland and Devlin, 1988).

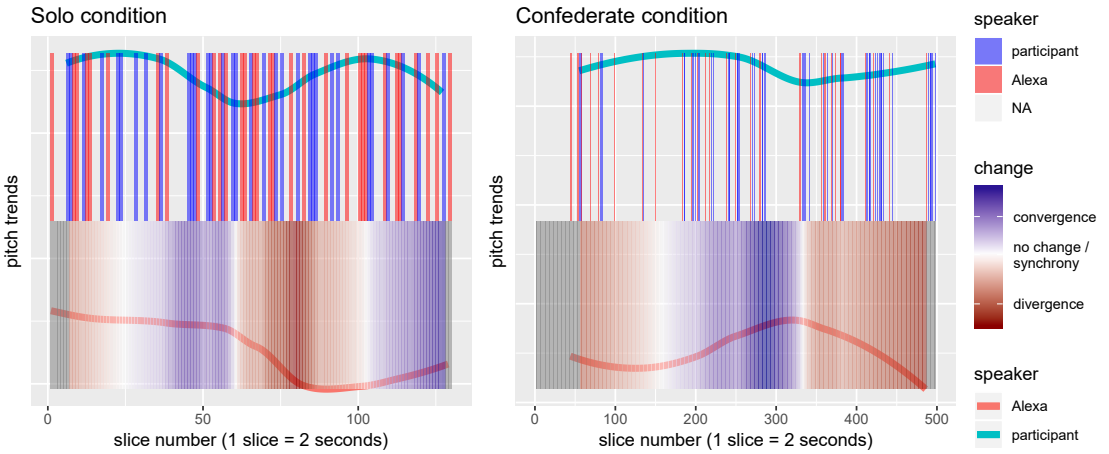


Figure 6.8: A comparison between f_0 changes in solo (left) and confederate (right) conditions. The horizontal lines show the smoothed trends of the participant (blue) and Alexa (red). Confederate turns and omitted segments (as per Section 6.2), are not colored (gray). The vertical bars in the upper halves represent the turns of the participant (blue) and Alexa (red). The color-scaled vertical bars at the bottom halves show the convergence (blue) or divergence (red) level of the participant as calculated by Equation 6.4.2. The darker the color, the greater the effect, with white indicating no change (or synchrony, in segments with both trends moving the same way).

Subsequently, the participant's contribution toward the accommodation is calculated by

$$accomm(participant)_t = change_t - \Delta_{t,t-1} S_{Alexa}. \tag{6.4.2}$$

The sum of the changes of each target feature produced by all participants in every interaction (i.e., a sex-task-condition-order combination) was calculated, resulting in a single value that represents the overall change per feature. A value greater than zero means that more convergence was observed, while a negative value points to more divergence. There were only two instances where this value was exactly zero, both for the AR feature. These instances were treated as cases of divergence, for their divergence spans were longer. Using this approach, only a few interactions had no feature convergence in them, and several had all three features showing convergence. However, a stricter threshold was applied, where a feature was considered as converging only if its overall accommodation value was higher than one standard deviation from its mean. Based on this criterion, all interactions were categorized by the number of features that showed

more convergence in them. Figure 6.9 summarizes this categorization for each factor. Each line represents a single interaction, and the strata it goes through form the factor combination of this interaction. The number of features that showed more convergence than divergence overall is marked by the color of the line. Some tendencies emerge from this categorization: first, in 35 % of the interactions, there was at least one feature that showed convergence, but in none of them did so all three features (though there were such cases with the more tolerant criterion). In seven interactions, two features showed convergence, two of which by male participants and five by females. In total, males converged in 5 % of all measurements and females in 7 %. Furthermore, of all converged features, 58 % occurred in the solo condition, compared to 42 % in the confederate condition. However, no substantial difference between the calendar and quiz tasks was found, with 49 % and 51 % of the cases, respectively. The same holds for the comparison between the two orders in which the tasks could be performed. These results support the addition of the confederate to the interaction as the factor for less convergence occurring in interactions.

6.5 Conclusion

The study in this chapter investigated vocal accommodation in human-human-computer interaction (HHCI). *Addressee* and *crowd* components of the study examined different aspects of simulated real-world use cases, in which a human user talks alternately with a computer-based interlocutor and another human interlocutor. The addressee component looked at differences between human-directed speech (HDS) and device-directed speech (DDS) within an interaction, while the crowd component focused on differences in DDS with and without the presence of the other human speaker. Distributional and temporal accommodation analyses were performed in both components to provide interaction-level and time-related insights.

Three features were analyzed in the study: fundamental frequency (f_0), intensity, and articulation rate (AR). The first two features show a greater degree of difference in both components than the third. Furthermore the changes were even greater in the crowd component, in which a human interlocutor was involved. This indicates that not only f_0 and intensity are more prone to variation than AR in general, but also that human interlocutors trigger these variations more. There several possible reasons for this difference. First, computer-based agents, such as Amazon Alexa used here, do not change

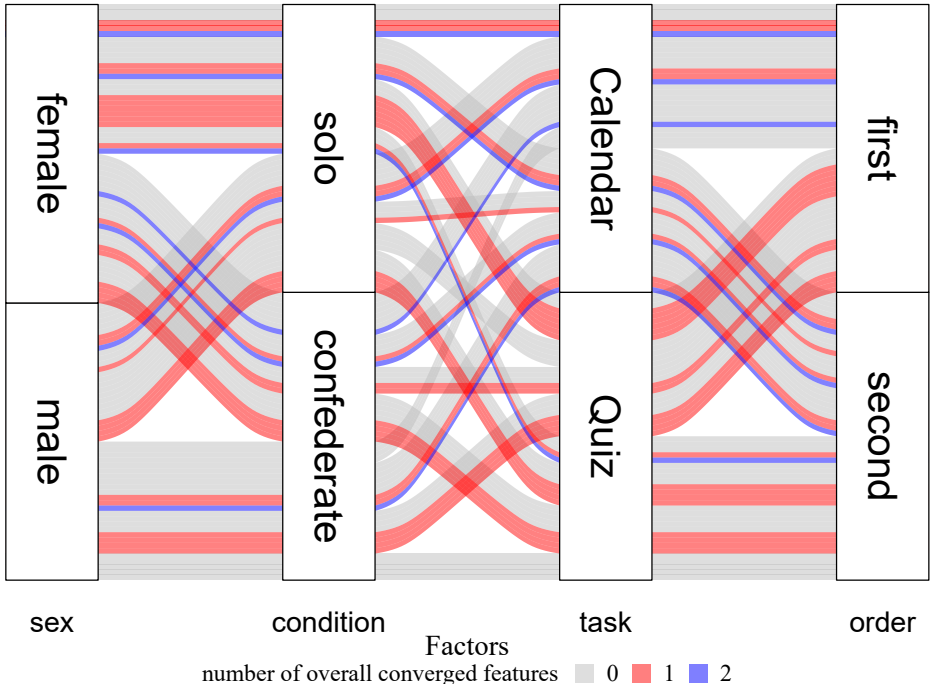


Figure 6.9: Overview of the relation between the factors `sex`, `condition`, `task`, and `order` and the number of features that showed more convergence in total across all interactions. Each line represents one interaction. The line colors stand for the number of target features that showed overall convergence in this interaction, from none (zero features, in gray), through one (red), and up to two (blue). For example, a blue line going through the strata sequence *female* → *solo* → *quiz* → *first* represents an interaction with a female participant performing the quiz task in solo condition first, in which the participant converged in two out of the three target features.

their speech output in any way, regardless of variations in the user's speech input. As a social, mutual process, accommodation is more likely to be stronger if both interlocutors contribute to the overall effect. In this study, the process can only be mutual in HDS, where another human is involved. Nevertheless, accommodation indeed occurred in DDS as well, both with and without the presence of the confederate. Secondly, due to this spontaneous nature of accommodation in HHI, the participants might have automatically spoken more dynamically toward the confederate, due to the inherently expected dynamic variations is in human interlocutor. Finally, VAs (and voice-activated devices as a whole) are, for the most part, still not performing well enough for people to speak to them as fluently and naturally as with humans. This results in a different manner of talking to machines. For example, users are more likely to articulate whole sentences,

typically without reformulating, when requested to repeat their utterance by the device, whereas human interlocutors are capable of requesting and conveying corrections with shorter segments using intonation and other means. Furthermore, although generally more seldom vary, users tend to reduce their AR when repeating their utterances to the device, or when trying to be more clear. This resembles the way people talk to children when they want to be more clear. However, due to the way automatic speech recognition (ASR) systems are trained, more often than not this achieves the opposite result. While participant did not change their AR much toward Alexa, they did show accommodation in f_0. This can be explained by the natural difference in male and female f_0 ranges, which leaves room for accommodation for participants of both sexes (Alexa used a female voice while the confederate was always male). In that case, the results shown here point to the fact that the participants generally treated Alexa as a human interlocutor with regards to f_0 behavior, as opposed to talking to Alexa using a more monotonous f_0. This happened despite Alexa not varying her f_0 beyond the sentence-level intonation. A similar effect was found for intensity. Since the device and the confederate were squally distanced from the participants, there was no apparent reason for the participants to speak more loudly with either interlocutor. Therefore, an explanation of the tendency to speak more loudly to the device may come from the intuition that a computer-based system has a harder time understanding human speech and therefore needs a clearer signal (also using lower AR, as explained above). This stands in line with the interpretation that humans sometimes treat voice-activated devices as humans who need a hyper-articulated speech signal to understand, like toddlers or language learners[16]. Another explanation may be the illusion that Alexa feels more distant than the human interlocutor because she is not an embodied agent (cf. Staum Casasanto et al., 2010; Gijssels et al., 2016, and see Section 3.2 for further details). Keeping in mind that humans strive to communicate as efficiently as possible, it seems like changing these features helped the participants – or at least felt like they did – to interact better with Alexa. Regardless of whether this roots from the unconscious attempt to treat the device as a social actor in the

[16]In a sense, this analogy is correct, as the devices – and specifically their ASR components – indeed learn to process speech signals. Despite that, this process is different than the way babies acquire spoken language, and therefore such projection on the speech style does not help the device improving its understanding of the user. Moreover, slower or disfluent speech may actually hinder the systems' understanding, as they are typically not trained on this kind of speech.

conversation (Nass et al., 1994; Nass and Moon, 2000) or the fact that accommodation occurs, even if to a lesser extent, even when it's not mutual, the effect still took place. This is, however, not the case with AR, which shows a lower percentage of significant differences. This indicates that AR does not tend to vary as much as f_0 and intensity, which stands in line with other studies, like Schweitzer and Lewandowski (2013). Slower, more carefully articulated speech, occurs less often in regular speech than louder speech or higher pitch. Such enhanced articulation not only takes longer to produce, but also requires more effort, making it a less preferred way to communicate, unless necessary. In this somewhat formal experimental setting, participants are likely to speak more slowly than usual, and the motivation to complete the task in a short time encourages them not to speak even more slowly. This supports the hypothesis that extra slow speech would only be used when necessary, e.g., when a repetition is required due to a recognition error on the system's side. Once the misunderstanding was resolved, participants' went back to their original AR. These local changes may suggest that changes in AR are done more consciously than in other phonetic features.

The results presented in Tables 6.1 and 6.2 show for all three features (a) that the distributions differed more when the participants first interacted with the device alone, and (b) that more convergence was aggregated in the task that was performed first. This demonstrates the influences of the `order` factor. Furthermore, the factors `sex` and `task` indicate that female participants showed a slightly higher amount of convergence in total than male participants, and that the performed task did not play role the increase or decrease of convergence. The first speech input a participant encounters may cause a priming effect that, together with the natural tendency to converge to an interlocutor, results in a greater change in interactions that occur first. However, the interchangeability of input (here, both HHI and HCI) seems to hinder the ability of the participants to converge to Alexa. One explanation for this may be that it is more natural for humans to accommodate to other humans, so once another human is involved, the accommodation towards the computer-based interlocutor is annulled. Another possible explanation is that due to the multiple interlocutors, the participants do not have a steady target to accommodate toward, which leads to a weakened convergence effect. This is confirmed by the higher number of convergence instances in the solo condition compared to the confederate condition in both the temporal and distributional analyses. This could be explained as a reaction to a more stable vocal target (especially since

Alexa generally doesn't change her voice much) than when alternating between two interlocutors. Since HCI still lacks the mutuality of accommodation effects, the question arises whether these tendencies would be stronger in interactions with a single human versus interactions with two human speakers simultaneously. The higher number of convergence instances by female participants may be ascribed to the VA using a voice of the same sex and could be further investigated by using a VA with a male voice.

III

Modeling

Chapter 7

Computational Model

For a computer to express accommodative speech, a machine-understandable description of the process is required. This chapter introduces a model of the accommodation process in humans, based on the finding in the human-human and human-computer studies presented in previous chapters. It's motivated by empirical data observed in these studies and aims to resemble the process that occurs in humans. The human-centric insiprations and the parameters representing them are discussed and demonstrated.

7.1 From HHI to HCI

The experiments presented in Part II show humans' accommodation behaviors in different human-human interaction (HHI) and human-computer interaction (HCI) settings. Complementary to those, Section 3.3.3 discusses ways to represent the various levels of accommodative behaviors in computers. Since behavioral changes, as explained by communication accommodation theory (CAT) (see Section 2.1), happen naturally and often unconsciously in HHI, it is not trivial to transfer them to computers, as they need defined rules and numeric representation to process. Numeric values can be achieved by applying some measuring technique appropriate for each feature, like formant values for vowel quality or frequency for fundamental frequency (f_0). Rules for different behaviors, however, cannot be directly constructed, due to the intuitive nature of this inherent phenomenon, but can instead be inferred from observed HHI data. For example, the results of the study described in Chapter 5 reveal a high degree of variation across participants (Section 5.3.2). While this is expected, further examination of these variations uncovers some general properties that can be taken as a basis for characterizing the different changes. The link between the measured raw values and systematic rules should be a machine-applicable equivalence of the accommodation process in humans. Such parallelisms would need to model, for instance, the way humans percept accommodation-prone sounds, how they interact with previous instances and the current internal state of the sound in question, and the cognitive process of the phonetic changes. Therefore, such a model should include the perception of the phonetic features, representation of the state change of each feature in memory, and the realization of the changes during a conversation. Since these steps are dependent on each other, the computational model presented here is depicted as a pipeline, in which each of its steps stands for one of the sub-processes mentioned above.

7.2 Pipeline representation

As accommodation happens automatically and seamlessly in human interlocutors' cognition, it is hard to say what are the exact steps this process comprises. Some key properties stand out when examining many occurrences in different interactions. The pipeline presented here aims to capture these properties and offer a defined process of describing the changes. The advantages of representing this process as a pipeline (as op-

posed to, e.g., a one-step, end-to-end conversion) is that each facet of the process can be interpreted and controlled separately (as demonstrated in Section 9.1.1 and Figure 7.1). This enables combining and experimenting with different methods and implementations for each step while maintaining their independence of each other. The ability to substitute a single step's logic without influencing ther others grants the freedom to experiment with different methods without losing the core principles of the accommodation process. For example, the calculation method in the *update* step can be changed or even replaced by a statistical model (like the one introduced in Chapter 8). Note that although only speech-related features are discussed here, this pipeline could be used for describing changes in other types of features and modalities, such as lexical choices, linguistic complexity, eye gaze, emotional state, etc.

The proposed pipeline representation consists of the following five main steps, which are explained in detail in Sections 7.2.1 to 7.2.5:

detect hear and identify a feature that can be changed, corresponding to a human ear's ability to notice such variations;

filter decide whether the encountered feature's instance appears in a way that can trigger change, capturing the inherent detection of phonological context and rules in the language;

store add the instance to the exemplar pool of this feature, representing the inventory that builds the internal representation of the feature;

update update the feature's state, standing for the "recalculation" of the way this feature is perceived by the speaker; and

assign apply and potentially limit the updated state, expressing the change in a speaker's production of this feature with the individual preference as to how far to go with a change.

The output of each step is the direct input for the next, except for cases where the execution is discontinued due to an unmet condition (see Figure 7.1).

7.2.1 Detect

This step stands for the human ability to identify phonemes in speech and analyze the way they are realized. The input to this first step in the pipeline is the raw speech signal of the speaker and its output is a sequence of realizations that may contain phonetic changes. The automatic speech recognition (ASR) component of the system is

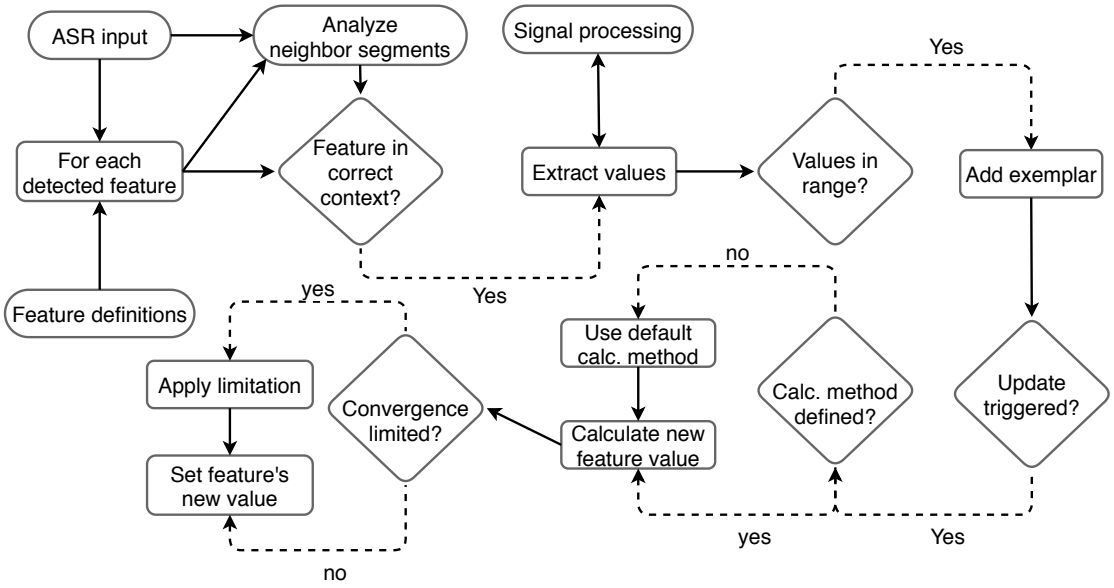


Figure 7.1: Overview of the vocal accommodation pipeline. Rectangle nodes represent steps where an action is performed, round rectangles are inputs (either external or from the system), and diamond-shaped nodes stand for decision points. Nodes without a "no" edge indicate termination of the process at that point if their condition is not met (and therefore no accommodation takes place). The pipeline is successfully completed only once the "Set feature's new value" node is reached.

responsible for detecting these realizations and their timestamps in the signal. With this information, various methods can be used to define and measure target features to take into consideration and pass forward. Section 9.1.1.1 shows an example of such a definition and how it is used in a spoken dialogue system (SDS).

An interlocutor cannot accommodate to features that are not present in a speaker's speech. For changes on any level to happen, some pre-defined feature that is prone to change needs to be present and detected in the input speech stream. Moreover, for the changes to register as a variation of a feature, the realization produced by the speaker must be prominent and distinctive enough to be perceived by the listener. In the case of computers, that means a way to measure the difference between realizations and classify their distance from one another. This difference can be categorical or continual, depending on the feature. In addition, not only the features which introduce meaningful difference are language- and culture-dependent, but they might also differ based on the specific situation in which the interaction takes place. For example, a segmental feature of a language where a phoneme can be realized in two alternative ways (as the

allophonic alteration explained in Section 5.3.1.1) will probably be ignored by a speaker of a language where only one of these vowels exist in its repository. In this case, the two vowels will simply be mentally merged into one, without causing any difficulties with comprehension. Suprasegmental features, like f_0 contour and articulation rate (AR), occur globally across the speech signal and are more often cross-lingual. Still, for a computer to be able to detect and track changes in those features, a way to measure and compare them is required.

7.2.2 Filter

This step corresponds the human internal, often unconscious, linguistic knowledge, and how it is used to decide which detected realizations are valid new instances that will be stored in memory. Its input is instances of a defined target feature detected in the previous step and it outputs those instances that should be stored as exemplars of their respective features. Section 9.1.1.2 demonstrates how a filter is applied based on a phonological rule and a feature definition with a target phoneme.

A target phoneme serves as an anchor for a rule that aims to capture a phonetic feature or a more evolved phonological rule. For example, the German phonological rule of [ə] elision at word-final *-en* (as described by Equation 5.3.1) can be captured by the phoneme sequence /Cən/, where C represents a consonant (although in practice only a subset of the German consonants inventory can be placed at this position). The anchor phoneme is [ə], since this is the segment that is subject to the change, namely the length – or complete absence – of it. therefore, for this phenomenon, the target phoneme would be [ə] and the measured feature would be segment length. Another target feature described in Section 5.3.1.1 is [eː] vs. [ɛː] realization of the mid-word grapheme *ä*, which is captured by the target anchor phoneme [ɛ] in a non-final position of a word. Any defined feature goes through two filters. First, the phonological context of the detected sequence is matched against the one defined for the target feature; and secondly, the defined accepted value range that would make it an acceptable instance of that phenomenon, like [ə] length between 0 ms and 60 ms or appropriate F_1 and F_2 values for the [eː] and [ɛː] vowels in the aforementioned features. After applying these two filters, only those instances that truthfully capture the desired phonetic phenomenon are kept. Another purpose of this step is to provide a way to integrate phonetic expertise to be used in the process. This helps not only to be more accurate regarding language-

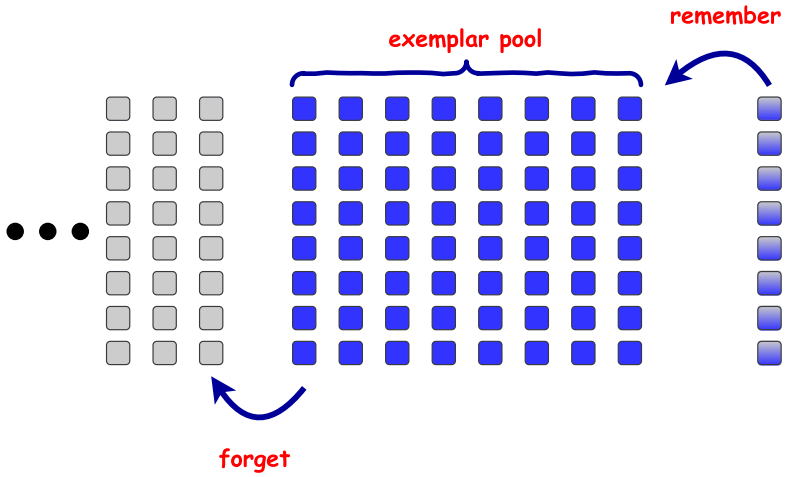


Figure 7.2: Illustration of an exemplar pool. Each exemplar is represented by a column of squares (each representing a numeric value). A new exemplar is added to the feature's pool when encountered. Old exemplars are removed when the pool is full. Exemplars currently in the pool are taken into account when the realization of the feature is determined.

specific knowledge, but also to prevent ASR errors to propagate further in the pipeline.

7.2.3 Store

This step represents the mental phonetic memory of a speaker, here referred to as a *pool* for a computer-based interlocutor. The input to it is a feature's exemplars that passed the filtering step and its output is used for updating the feature's representation. An illustration of an exemplar pool is shown in Figure 7.2 and an implementation of it is described in Section 9.1.1.3.

After an instance of a feature is detected and validated, it needs to be registered as an exemplar of the feature it is associated with. This stands in parallel to the way such exemplars (and in other contexts also words, meanings, etc.) are mentally stored in the human's short- and long-term memories. These accumulated exemplars of a feature determine how a speaker perceives it and shape its production when used in speech. One of the complexities of modeling such internal representation is the interleaving influences of both long-term and short-term memory. In spoken interaction, the long-term memory may define the typical productions of a user, while the short-term memory is used and changed within a single conversation. Since this model aims to describe accommo-

dation occurring within the scope of a single, isolated interaction (even if a long one), only the storage of exemplars encountered in this interaction is explicitly addressed in it. However, long-term effects may be implicitly achieved by retaining values between interactions. This step also defines how new exemplars are added, stored, and removed (cf. Figure 7.2). Each feature has its own exemplar pool to which newly encountered exemplars are "memorized", and each exemplar is a vector of the values measured for the target feature, e.g., formant values). The pool functions in a first-in-first-out fashion, fitting the temporally linear progression of spoken interaction. An exemplar is represented by a vector with cardinality n, where n is the number of dimensions required for describing this feature. Whenever an exemplar of the feature is encountered, a new exemplar is added to the pool. The size of the pool determines the memory capacity, i.e., for how long exemplars are remembered during the interaction. If the pool is already at full capacity, the oldest exemplar is "forgotten" when a new exemplar is added. Ultimately, a pool of a feature can be used to determine which exemplars are still affecting the speaker's mental state of a feature. As the order of the added exemplars is kept, it can be taken into account as well when determining each exemplar's weight, just like recent turns are more likely to influence the current utterance than turns from the beginning of the conversation.

7.2.4 Update

This step incorporates the process of changing the mental state of a feature based on its accumulated exemplars. The input to it is the current state of a feature and it outputs a new value for it.

The core of the accommodation process is the change in a feature's state. Many factors may influence this change, both internal and external. The two main considerations in this step are one of each, namely the desired accommodation behavior and the exemplars collected from the user's speech input. The latter is covered by the "store" step, and the former is defined using adjustable parameters that correspond to accommodation properties in humans (see Table 7.1). For example, how prone is the speaker to be influenced by others' speech and how easily should the change be triggered. The sensitivity can be constant or vary based, e.g., on how close are the speakers' productions to begin with. A trigger might be exemplar-based, i.e., after a certain number of new exemplars were added, or time-based, i.e., every time a certain amount of turns

had passed. Another means to shape the behavior is the way the new state is computed based on the exemplars. For example, newer exemplars or exemplars with greater distance from the current state may influence the change more. Moreover, the general tendency to accommodate is defined in this step, e.g., converging or diverging from the user's speech, which is determined, among others, by the application and desired behavior. A computer-assisted pronunciation training (CAPT) system would probably not aim to align its speech to the user's, but rather diverge from it as a way to provide auditory feedback. This computation can use simple mathematical operations (as demonstrated in Section 9.1.1.4) or more involved data-driven statistical methods (as the one in Chapter 8).

7.2.5 Assign

This step mediates between the new state of a feature and its use in the system's speech output. The input to it is the newly calculated state of a feature and it outputs a potentially altered version of this state to be used by the text-to-speech (TTS) component.

This final step of the pipeline is responsible for assigning the features' representations to the speech production of the system as an additional input to the system's TTS component. For a TTS module that can directly control the speech output (as part of the model itself or on the outputted waveform, as discussed in Section 9.2.2), this additional information can be used to manipulate the target features in a way that expresses the accommodative behavior of the system. This closes the circle from a target feature produced by the user and up to the change it triggered in the system when it speaks back. Since that means the user will now hear certain vocal characteristics that are based on their own speech, it is important to avoid a situation where the user feels imitated – or even mocked – by the systems. This is an important issue that has not been considered in previous work. To that end, this step introduces a limitation mechanism that limits the values given to the TTS component. The values are re-evaluated if some threshold is bypassed (see Equation 9.1.8), to avoid such imitation from the system's side. This mechanism also helps to prevent the system from diverging too sharply from the user. For example, a CAPT system might demotivate or frustrate the user if its speech is consistently considerably different from the user's. From a human's perspective, this step corresponds to the natural degree to which a speakers would change their speech while talking to others. As shown in Chapters 5 and 6, this varies from feature to feature,

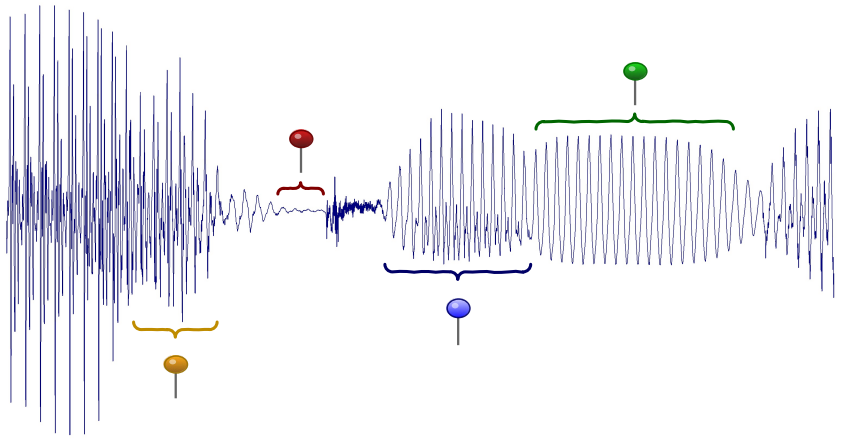


Figure 7.3: Illustration of a manipulated output audio waveform. Each colored pin marks a phonetic features captured and processed by the pipeline.

and hence this parameter is set for each feature individually (see Table 7.1). For this to work, it is important that the features are sufficiently distinguishable and clearly defined by the pipeline, so that the specified modification in the system's speech output can be properly applied in – and only in – the correct places, as illustrated in Figure 7.3.

7.3 Parameters

Several parameters are introduced into the pipeline described in Section 7.2 to grant degrees of freedom in shaping the accommodation behavior. These parameters link between the theoretical, schematic model and its integration into a SDS, as demonstrated in Chapter 9. They are also the key to experimentation with different settings and scenarios for different applications. The model's parameters are summarized in Table 7.1 (and cf. Raveh et al., 2017b).

As shown in Section 5.3, not all participants showed the same *sensitivity* toward changes in the stimuli. Here, sensitivity refers to the degree of overall change toward external speech input. Additionally, when one does converge, the sensitivity to changes (i.e., the "amount of differentiation") toward every single stimulus might differ as well. These two aspects are jointly controlled by the *convergence rate*, which represents the balance between the current and heard speech when calculating the accommodation outcome. Generally, low convergence rate leads to a slow (and potentially unnoticeable) change, while a high rate would lead to sharper changes and that may overshoot the

Table 7.1: Computational model's parameters in their order of use. The colors mark parameters associates with the **detect**, **filter**, **store**, **update**, and **assign** steps.

Parameter	Description	Value
target phoneme*	the phoneme that triggers the feature's pipeline	a phoneme symbol
phonetic context*	the environment in which the feature instance is accepted	regex containing the of phoneme symbols containing target phoneme
allowed range*	the value range(s) in which new instances are accepted	two numeric values (min and max) per feature dimension
exemplar pool size	maximum number of exemplars in memory at a time, oldest exemplar removed when full	positive integer
update frequency	how frequently a feature's value is recalculated, controlling the accommodation pace	non-negative integer; 0 for manual update
calculation method*	the manner in which the pool value is calculated based on the values and order of the exemplars in pool	any $\mathbb{R}^{n\times m} \longrightarrow \mathbb{R}^m$ function; either implemented directly in code or sent to an external statistical model
convergence rate	weight of the exemplar pool when updating the feature's state, controlling the impact of external input on the speaker's features states	real value; typically $n \in (0,1) \subset \mathbb{R}$; 0 for ignoring the pool; > 1 for over-weighting the pool value
convergence limit*	the maximum degree of convergence allowed for the feature with respect to the input instances	real value; $n \in (0,1] \subset \mathbb{R}$; 1 (100 %) for no limitation

** denotes parameters that are defined individually for each feature*

target, as demonstrated in Table 10.2 and Figure 10.5. To simulate the case where a speaker is not influenced by external speech input (the exemplars in the pool) at all, this rate can be set to zero. In that case, the model will ignore the other interlocutor's speech and stick to the current speech style. Another difference found among the participants was the total overall convergence degree toward the stimuli, i.e., where does the convergence process stop. This is monitored by the *convergence limit*, which defines the maximally allowed degree of similarity between the interlocutors. When set to 1 (100 %), the model is allowed to change up to 100 % toward the other interlocutor (complete convergence); when set to 0.8, up to 80 %, and so on. The parameter ensures that the model does not simply imitate the user's input, which is the approach often found in such system nowadays. By limiting the change, the accommodation process is more gradual and restrained, avoiding peaks and abrupt changes.

Parameters defining the adaptation itself are not enough. To properly model an accommodative behavior, some aspects that are not directly related to the speech output are required as well. The accommodation process relies on the recent instances (*exemplars*) of a speech sound. How many exemplars are taken into account when the feature's state is updated depends on the interlocutor's mental memory of that sound. This internal memory is a complex mechanism (Baddeley, 2003), which is simplified here into a single parameter, namely the *exemplar pool size*, which determines the number of exemplars the interlocutor currently remembers. This exemplar history is managed on a first-in-first-out basis, so that the *order* in which the exemplars were acquired is kept as well and can be used for weighting their influence. This pool size can be tweaked to match the scenario and the expected interaction length. The *tendency to converge* toward other interlocutors also differs from speaker to speaker. This likelihood is controlled by a parameter the parameter *convergence rate*. After an exemplar is added to a feature's pool, an update of the feature's value may be triggered. Whether and how often this happens is determined by the *update frequency*. When set to 1, an update will occur every time an exemplar is added; if set to 2, every other exemplar, and so on. When set to 0, however, updates will only take place when explicitly requested, e.g., after a pre-determined number of turns or a fixed amount of time. This can be useful when all features are to be updated at the same time, regardless of how many exemplars have been accumulated for each of them. Increasing the update interval means that each update will be affected by a higher number of new exemplars, which might result in a

smoother converging process, depending on the calculation method used (see below). Additionally, a longer update interval also means that accommodation will generally take longer, since the model's features are not being updated as frequently. This is fitting for systems with which the user is expected to have long interactions.

Not only the frequency of updates plays a roll in the process, but also the manner in which the update is performed. This manner is determined by the *calculation method*. Since the features in the model are represented by vectors, any function that takes a matrix as input and outputs a vector as output can be used, as demonstrated in Section 9.1.1.4. The method can be either statistical or deterministic, e.g., simply averaging the exemplars, but methods that can take order into account, like decaying average, might yield more realistic accommodative behaviors. A different calculation method can be assigned to each feature, which can help to account for acoustic or psycholinguistic constraints. This can also be influenced by the setting the system is purposed for, like experimental, exploratory, data collection, etc.

For each feature, a *target phoneme* is defined, which serves two purposes: First, it tells the ASR component which phoneme is associated with this feature, to later forward it for further analysis. Secondly, it is used in the *phonetic context* to filter instances of the phoneme that should not be associated with the feature. The context is the environment in which the target phoneme should be found in the ASR output sequence for the instance to be considered[17]. For suprasegmental features – or any other feature that is not bound to a specific phoneme or context – the target phoneme can remain empty, so that the phonetic context would match any sequence. The second parameter used to put constrains on the detected phones is the *allowed range*, which defines the minimum and maximum acceptable values for the feature. This parameter is important to obtain clean and sensible exemplars, as it introduces restrictions based on phonetic knowledge (e.g., reasonable f_0 values for a human speaker), which at the same time also help to prevent ASR errors from meddling with the exemplars sent to the pool. Since these values are feature-dependent, this parameter is set for each feature individually.

[17]This requires an ASR engine that returns such a sequence in addition to textual output. It is therefore crucial that the phoneme symbol set used in the model and by the ASR is the same and unambiguous, specifically when used in a regular expression. This is the only part of the pipeline that is language-dependent (or rather symbol-dependent) Using non-ASCII symbol sets, like IPA, may solve many of these issues, but is not recommended since ASR engines rarely use those and also due to other technical reasons.

Chapter 8

Probabilistic Variational Model

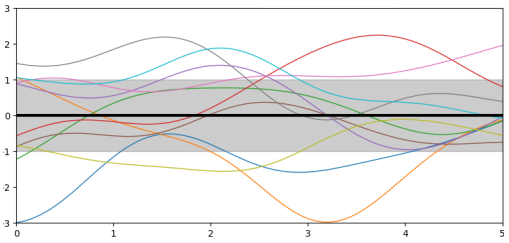
GRANTING a system an accommodative behavior beyond a direct response to the user's input is a major and understudied challenge in HCI. In live HHIs, the behavior of a speaker varies both within and across conversations. This chapter presents a statistical approach that generates variational behaviors, which can be integrated into a spoken dialogue system to offer non-deterministic accommodative outputs.

The statistical approach presented in this chapter offers a different way to model and simulate accommodative behaviors. It determines the system's next output value based on probabilities calculated from a dataset, unlike the method introduced in Chapter 7 where the output value is determined based on pre-defined parameters. While the latter achieves *responsiveness* in a spoken dialogue system (SDS) per the definition in Section 3.3.3, the former grants *profiles* and *variability*. This is done using two ways to describe accommodation in a conversation. The first is a continuous representation, where the speakers' productions are treated as time series and interpolated. The chosen interpolation lines estimates the speakers' productions for the whole conversation, as explained in Section 8.1. The second representation uses accommodation categories to describe the nature of the mutual changes observed in the speakers' productions (e.g., convergence or divergence). As shown in Section 8.2, these categories allow to represent accommodation more abstractly and give meaning to the productions' raw values. Using this unified representations, n-gram probabilities of these categories can be calculated to predict the next accommodation event in a conversation. Section 8.3 shows how these two representations are combined into one statistical model. The overall type of change is generated by the n-gram probabilities and the specific variational value is determined by selecting an interpolation line based on the already observed productions. This process is demonstrated by clustering speaker behaviors from a dataset to generate different system outputs for a given user input.

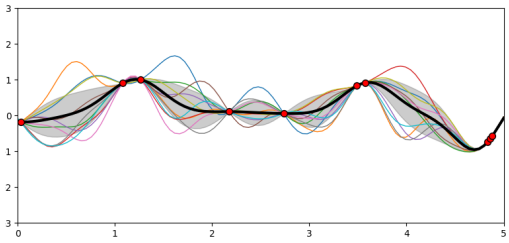
8.1 Time series representation with Gaussian processes

The approach presented in this chapter capitalizes on the temporal nature of spoken interaction, with emphasis on the evolution of mutual accommodation over the course of an interaction, which expands upon the temporal analyses performed in Chapters 4 and 6. As in those chapters, features' values are chronologically sampled across equal time intervals, and are therefore treated as *time series*. By extension, accommodation can be viewed as time series ass well. This is motivated by the arguments explained in Section 6.4.2, vis. that representing mutual changes by merely a few points throughout an interaction or directly comparing its beginning state to its end state is not very informative and draws a rather simplistic picture (see Section 2.2.2 and Figure 2.2 for details and examples).

This variability in this approach is achieved by fitting a *GP* to each speaker. GPes



(a) RBF kernel prior ($lengthscale = 1$)



(b) RBF kernel posterior ($lengthscale = 0.279$)

Figure 8.1:[18] Prior and posterior distributions of an RBF kernel with mean zero, resulted in a Gaussian process $\mathcal{GP}(0(\vec{x}), \Sigma(\vec{x}))$. Each color line stands for a drawing (prediction) from the prior and posterior distributions, and the thicker black line shows the overall mean of the distributions. The red circles are the known datapoints on which the kernel was optimized to fit, and the gray areas mark the 95 % confidence intervals above and below the overall mean. The length scale parameter (in parentheses) determines the length of the "wiggles" of the functions

are stochastic processes with a multivariate normal distribution for each random variable. A GP provides the joint distribution for infinitely many random variables, i.e., a *distribution over functions* that match the given evidence. Specifically, GPes are used here for *kriging* (Section 8.1.2), an interpolation technique used for time series interpolation and prediction. The utilization of GPes provides not only continuous likelihood line for the observed features, but also an infinite number of non-deterministic alternatives – the variations – to generate vocal behaviors by randomly sampling from a GP. The generation process presented here can be combined with the aforementioned computational model to harness the benefits of both, as discussed in Section 9.1.2.

8.1.1 Kernel building and tuning

Kernels (or *covariance functions*) are a key component in GPes, as they define the similarity between the GP's random variables. They define the the covariance $k(x, x')$ between each pair of observed values x and x', so that $k(\cdot, \cdot)$ determines how similar the outputs y_* and y'_* will be. Formally, a covariance function can be described as $\mathcal{K}(u, v) = \phi(u) \cdot \phi(v)$, where $\phi(\cdot)$ is a function that maps the input vectors into a transformed

[18] Adapted from https://scikit-learn.org/stable/_images/sphx_glr_plot_gpr_prior_posterior_001.png

feature space. Which function to use is a key consideration when using GPes, as it determines the behavior of the sampled functions and the nature of the predictions it will be making. A kernel's parameters are optimized to achieve functions that better fit the data. Since accommodation analyses deals with the *difference* between production values (as opposed to the values themselves), stationary kernels are more suitable for fitting GP for them, as they are shaped by the distances between each pair of datapoints rather than their absolute values. A kernel may also be composed of multiple other kernels, to capture a combination of characteristics. This is done either by multiplication or addition of these kernels. Multiplication-based kernels are maximized when all of its kernel factors yield high values, whereas addition-based kernels are maximized when any of their addend kernels yield a high value. An additive kernel with constant, radial basis function (RBF), and noise terms is used here (see Equations 8.1.1 to 8.1.3 below). The RBF term determines the general shape of the curve (see example in Figure 8.1), the constant term enables shifting of the curve if necessary, and the noise term adds degrees of freedom in case the curve cannot completely fit the input signal.

The definitions of the individual kernels are as follows:

Constant kernel – is a simple kernel that assigns the same value for all input pairs. Since by itself it does not offer a lot of characteristic to the covariance function, it is usually used in combination with other kernels, where it scales the magnitude of the other factors, or as part of a sum kernel, in which it modifies the mean of the Gaussian process. It has a single parameter, the constant value, and it is defined as

$$k_{constant}(\{C\}, x_1, x_2) = C \forall x_1, x_2, \tag{8.1.1}$$

where C is the constant value parameter.

Noise kernel – is a kernel used for capturing unexplained variation in the data. It is typically based on the constant kernel as part of a sum kernel, in which it explains the noise component of a signal. In this context, the constant parameter is tuned to estimate the noise level in the interlocutor's mutual change (including distortions

caused recognition errors). This is determined by

$$k_{noise}(\{noise\_level\}, x_1, x_2) = \begin{cases} C_{noise\_level}, & if \quad x_1 = x_2 \\ 0, & otherwise, \end{cases} \tag{8.1.2}$$

where *noise_level* equals to the variance of the noise found in the input signal.

Radial-basis function (RBF) kernel – also known as *squared exponential kernel*, this kernel is a stationary kernel with one parameter, *lengthscale* $\ell > 0$. This kernel typically results in generally smoothed functions, with the lengthscale being associated with the long-term smoothness and degree of variability on the time dimension. The RBF kernel is defined as

$$k_{RBF}(\{\ell\}, x, x') = \sigma^2 exp\left(\frac{\|x_1 - x_2\|_d^2}{2\ell^2}\right), \tag{8.1.3}$$

where $\|x_1 - x_2\|$ is the Euclidean distance between two d-dimensional input points and σ^2 is the data variance. Figure 8.1 shows prior and posterior examples of the RBF kernel.

8.1.2 Data interpolation

As also motivated in Sections 2.2.2 and 6.4.2, artificially splitting interactions into a fixed, pre-determined number of parts to measure accommodation results in a limited view on the accommodation events due to sparsity of observation. To overcome that, the mutual changes should be evaluated continuously throughout the interaction instead of by point-to-point comparisons, where the temporal gaps between datapoints might be greatly unbalanced. This requires some interpolation method to achieve more general trends based on the observed productions. One way of achieving that is using some smoothing algorithms, like locally estimated scatterplot smoothing (LOESS) in Figure 6.7, which is adequate for gaining a smooth estimation of a speaker's overall performance. A similar approach is used in Galvez et al. (2020), in which the average values obtained by time-aligned moving average (TAMA; Kousidis et al., 2008; Kousidis and Dorran, 2009) define the behaviors. However, TAMA values may conceal turns with substantial changes

introduced by one of the speakers. A more evolved approach is presented here to describe a speaker's vocal behavior in a conversation as a distribution over functions that match the accumulated *evidence* from that speaker's productions.

Kriging (or *Gaussian process regression*) is an interpolation method that gives an optimally fitted and unbiased prediction of intermediate values. Since this method fits a function distribution over the data, it not only yields mathematically more likely values, but also provides a curve that describes the characteristics of the interpolated curved, as opposed to more naive methods like linear interpolation or smoothing spline. Another advantage of this method is that it offers a *distribution over functions* rather than specific values. Therefore, an infinite number of suitable functions can be sampled from one fitted kernel and their likelihood can be evaluated. Such samples are illustrated in Figure 8.1(b), where each line represents a mean regression prediction drawn from the posterior distribution based on the given datapoints. This method was applied to each interlocutor in the human-computer interaction (HCI) portion of the dataset presented in Section 6.2. It is described by

$$f_*(\vec{x}) = \mathcal{GP}(\mu_{\vec{x}}, \Sigma_*), \tag{8.1.4}$$

where $\mu_{\vec{x}}$ is the mean feature value of a single speaker and Σ_* is the fitted additive covariance function described in Section 8.1.1. It is important to note that the mean is not zeroed (as often done in GP regressions), to maintain the original input's mean for the subsequent steps. The kernel was initialized with the priors $C = 1$, $\ell = 1$, and no assumptions regarding the noise level ξ. The search boundaries for the RBF and noise components were $1 < \ell < 100$ and $1 \times 10^{-4} < \xi < 10$, respectively, with a maximum of six optimization iterations. The datapoints of the original series were grouped by the turn they belong to. Then, the average values of turns immediately before and after a floor change were taken as input datapoints for the GP. This results in evidence concentrated around input from the other interlocutor, with the objective to capture turning points that are more prone to accommodation. With the fitted kernel, a continuous prediction can be made for each speaker over the entire conversation timespan. Figure 8.2 shows an example of GP predictions for one of the conversations.

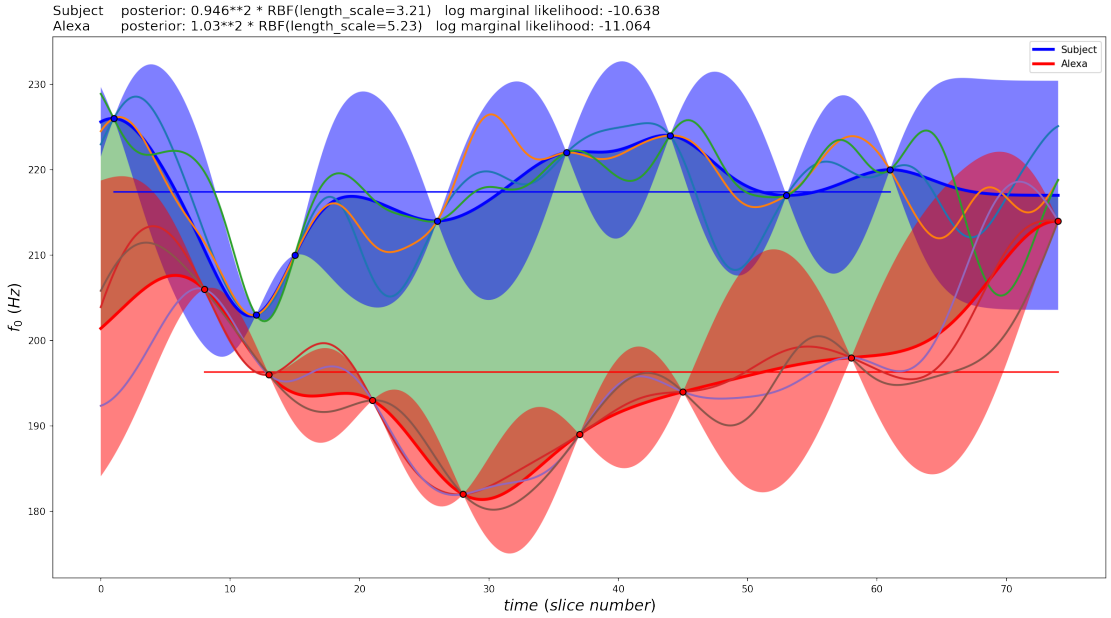


Figure 8.2: Gaussian process regression for an interaction of a participant with Alexa. The thick blue and red lines show the predictions' mean. The additional lines around the means are randomly drawn functions from the fitted kernel representing potential variational output. The colored areas around the means lines show the 95 % confident interval for the distributions of the same color. The straight horizontal lines indicate the overall mean of each speaker's productions. The posterior parameters and the log marginal likelihoods of the fitted distributions are stated at the top.

8.1.3 Marking degrees of change

Once a regression line is drawn for each speaker from their respective distributions, the differences between the speakers' productions can be measured. Due to the high temporal resolution used here, more fine-grained degrees of change over time can be calculated. These differences are calculated by the subtracting the trapped areas between the two regression lines (see Figure 8.2)

$$f_{diff} \equiv \Delta\vec{x}_* = \int_{\vec{x}_i}^{\vec{x}_j} \mu_{f_* participant} - \int_{\vec{x}_i}^{\vec{x}_j} \mu_{f_* alexa} \tag{8.1.5}$$

and the directional derivatives of the resulted delta line to measure the degree of change

$$\nabla\Delta\vec{x}'_* = \frac{d}{dx} f_{diff} \tag{8.1.6}$$

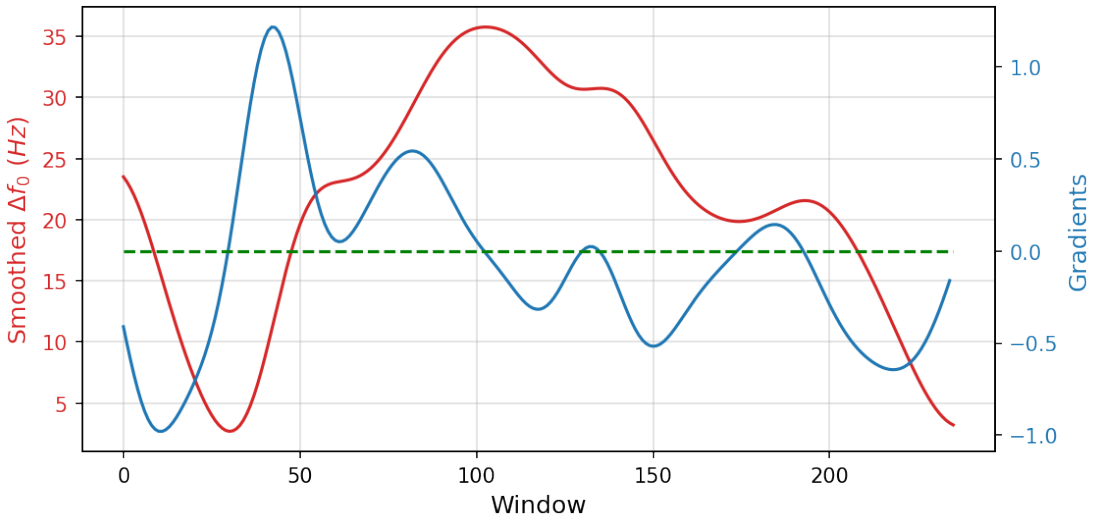
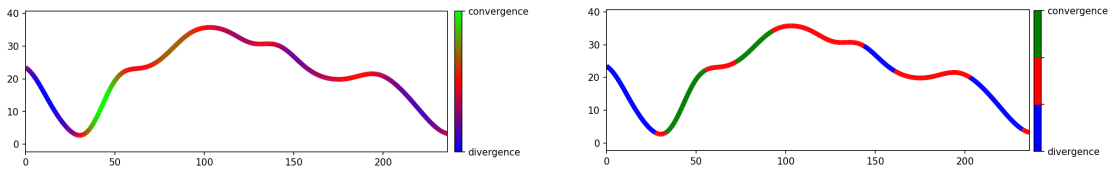


Figure 8.3: Continuous integral differences (red line) and their corresponding derivatives (blue line) of speakers' productions in a conversation. The dashed green line shows the zero-gradient, i.e., no-change threshold.



(a) Continuous scale (b) Discrete scale

Figure 8.4: Continuous and discrete color-coded scales for labeling degrees of change in a conversation.

along the delta line. Figure 8.3 illustrates these two measures on the same conversation from Figure 8.2 using the mean prediction of each speaker. For creating a generation process as described in Sections 8.2 and 8.3, the changes must be marked with predefined labels. To that end, the derivative values were translated into a continuum of change ranging from *divergence* to *convergence*. Based on this continuum, a discrete scale can be defined. The more categories this discrete scale offers, the more specific the behavior descriptions can be. Figure 8.4 shows this process for a discrete scale of three categories: divergence, no (major) change, and convergence.

8.2 N-gram representation for accommodation sequences

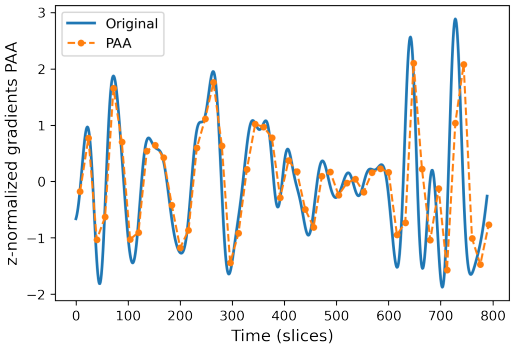
In order to generate accommodation sequences, a model that can iteratively emit labels based on the label history is needed (in contrast to, e.g., models with the Markov property, as in Bellman, 1957). An n-gram model is proposed here, which incorporates a portion of a sequence's history (*context*). The estimation of the element e at position i is calculated by the probability term $P(e_i \mid e_{i-n}, \ \dots \ , e_{i-1})$. N-gram models are traditionally used for describing sequences of linguistic units, like words in a language model (e.g., Niesler and Woodland, 1996), and are used in applications with sequential nature, like machine translation (Marino et al., 2006) and proteins identification (Xu et al., 2015). The n-grams model here describes sequences of *accommodation levels* in a conversation. These levels are represented by discretized values that are based on the degrees of change acquired in Section 8.1.3. After computing the n-grams probabilities of the level, this model can be used for generating new sequences. The resolution and variability of the model can be controlled by changing the n-grams' n and the number of levels used to distinguish between the different accommodation levels.

8.2.1 Dimensionality reduction and symbolic representation

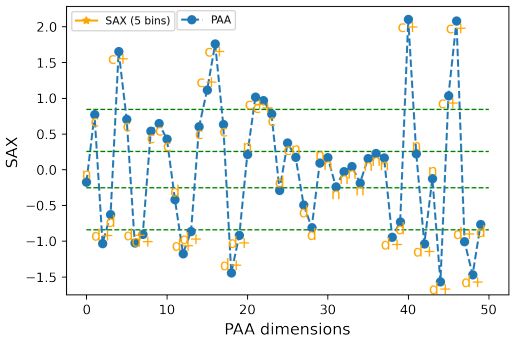
The time series gradient extraction process described in Section 8.1.3 is greatly high dimensional, even for short interactions. While this representation is useful for gaining a fine-grained overview of the data, it is not practical for various analysis techniques that do not benefit from (or are not designed to handle) such high-dimensional data. For instance, clustering algorithms, which need to iteratively compare between all points in a collection, scale better with lower dimensionality. The dimensionality of the data in question here is reduced using piecewise aggregate approximation (piecewise aggregate approximation; Keogh et al., 2001), which is a common dimensionality reduction technique for time series. Figure 8.5(a) shows the output of PAA on the gradients of an interactions. Each element $\vec{x}_i$ of the reduced vector is calculated by

$$\vec{x}_i = \frac{N}{n} \sum_{j=\frac{n}{N}(i-1)+1}^{i(\frac{n}{N})} S_j, \tag{8.2.1}$$

where n is the dimensionality of the original times series, $1 \leq N \leq n$ is the dimensionality



(a) PAA of the original time seires. The blue continuous line shows the z-normalied original time series of mutual change gradients in a conversation, which was extracted as described in Section 8.1.3. The organe circles show the PAA points created based on the continuous lines. The dashed orange line is the linear interpolation between the PAA points. Note that this lines is for illustration only and is not taken into account in the analysis.



(b) SAX of the piecewise aggregate approximation (PAA) values. The blue circles are the PAA points from Figure 8.5(a). The blue line visualizes the linearly interpolated trend of these points. The horizontal green dashed lines show the margins of the five bins that split the points discrete bins derived from a normal distribution. The organe labels ('d+', 'd', 'n', 'c', and 'c+') mark the classification of each point based on the bin it falls into.

Figure 8.5: Piecewise aggregate approximation (PAA) and symbolic aggregate approximation (SAX) of the time series representation of mutual change in a conversation.

of the output vector, and S_j is the j^{th} element of the original time series. PAA is suitable here, as the goal is to obtain vectors with fewer dimensions that still faithfully represent the data, and not, e.g., decompositions of the original data (cf. method survey in Keogh et al., 2001, pp. 271-275). Since the goal here is to compare *trends in change* within and across conversations rather than their absolute values, all gradient time series where z-normalized before applying the PAA. In the context of conversation analysis, the PAA's dimensional cardinality determines the "zoom" level, i.e., how fine-grained the representation is (the more points the more in detail the time series is described). Ultimately, PAA provides **continuous numeric values** that represent the overall shape of the original time series.

PAA is suitable for analyses of continuous numeric values. However, discrete values are required for symbolic n-gram sequences. For converting the continuous PAA values, the symbolic aggregate approximation (SAX; Lin et al., 2007) method was used. SAX assigns a string label based on a pre-defined number of bins, as shown in Figure 8.5(b). Such labels provide a more meaningful and compact representation. As explained by Apostolico et al. (2003), it is preferable to use a discretization technique that uses a

symbol set with equiprobability. To that end, the SAX discretization was done using bins based on the normal distribution of the z-normalized values. Five bins, representing accommodation levels, are used here for categorizing degrees of change, labeled 'd+' for *strong divergence*, 'd' for *divergence*, 'n' for *no (major) change*, 'c' for *convergence*, and 'c+' for *strong convergence*. This number of bins was found to adequately describe types of accommodation; three labels resulted in underspecified sequences where all degrees of convergence or divergence are labeled similarly, and seven or more labels did not provide substantial additional insights and often yielded sparse sequences. The motivation for choosing an odd number of labels is to have a neutral ("no-change") label and even number of labels for convergence and divergence, due to the assumption that they are equally likely to occur. However, an even number can be used as well, forcing each value to stand for either convergence or divergence while ignoring zero values. It is also possible to allocate more labels to convergence or divergence, if one of them is assumed to occur more in the data and a more fine-grained description of it is desired. Note that the term *synchrony* is avoided here for describing a steady distance between the speakers, as per its definition in Section 2.1.1 it entails additional properties regarding the individual change of each speaker. Ultimately, SAX provides **discrete textual labels**, which provide a discretized version of the original time series.

8.2.2 Sequence extraction and probability calculation

After applying SAX, a sequence of accommodation labels is obtained for each corresponding interaction. The count distribution of the labels was computed to examine their frequency. As expected, the symbol n had the highest frequency, about 2.5 times higher than the convergence and divergence labels c and d, whose frequency, in turn, was roughly four times higher than the frequency of the strong convergence and strong divergence labels $c+$ and $d+$. The same calculations were repeated for sub-sequences of the labels as they appear in the SAX sequences. The frequencies can be divided into three groups: matching the trend of the single-label distribution, repeated n labels were about three times more frequent than the second group, which consisted of most other sub-sequences that included n. Lastly, this group was followed by a long tail of less frequent sequences, starting from counts three to four times lower, which included the rest of the symbol combinations within a sub-sequence. Sub-sequences with many repeated $c+$ or $d+$ labels (i.e., sustained convergence/divergence) mostly appeared toward the

Table 8.1: Three examples of probabilistically generated accommodation level sequences. Each sequence consists of eight labels that were generated based on the initial context of the padding symbol p. The first line of each example shows the generated symbols as explained in Section 8.2.1 and their average variability (value between 0 and 1, higher number means more variability.). The second line lists the probability of each generated label given the context seen up to that point and the overall probability of generating this entire sequence. In the third line are the perplexity scores of the trigrams ending with the corresponding generated label given the context at the time of the generation. Note that the first two padding labels do not have any scores, since they are given as the initial context. Similarly, the first generated label does not have a perplexity score, as it can only be calculated once the sequence is longer than the one trigram.

p	p	n	n	n	c	n	d	d	c	variability:	0.187
—	—	0.44	0.46	0.55	0.18	0.32	0.29	0.25	0.35	probability:	1.63×10^{-4}
—	—	—	2.22	1.99	3.16	4.16	3.27	3.67	3.35	perplexity:	3.11
p	p	n	d	d	c	c+	d+	d+	c+	variability:	0.687
—	—	0.44	0.17	0.25	0.35	0.13	0.25	0.34	0.19	probability:	1.37×10^{-5}
—	—	—	3.67	4.88	3.35	4.69	5.62	3.46	3.96	perplexity:	4.23
p	p	c	c+	n	n	d+	c	n	n	variability:	0.375
—	—	0.30	0.25	0.25	0.16	00.04	00.20	0.22	0.41	probability:	2.16×10^{-6}
—	—	—	3.67	4.03	5.02	12.72	11.51	4.77	3.30	perplexity:	6.43

end of the tail. Increasingly smoother instances of the same overall distribution shape were found for all sub-sequence lengths from two and up to half the SAX sequence length (after which such frequencies become not as meaningful).

To calculate label probabilities, these sub-sequences were treated as n-grams with $n = 3$, i.e., trigrams. Similar to an n-gram language model, the size of the n-gram determines the amount of previously acquired evidence taken as context when calculating the probability of the a subsequent label. This fulfills a similar role as the *pool size* parameter of the computational model (see Section 7.3). In both cases, the goal is to consider the temporal evolution of the conversation for predicting its continuation. To account for conversation-initial sequences, the beginning of each symbol sequence was padded. The end of the sequence is not padded, since, unlike a traditional language model, the generation process here assumes that the user decides when to end the interaction and therefore does not attempt to predict it. This summed up to a collection of 2,700 trigrams from all interactions, which comprised 143 (92 %) out of the $5^3 + 1 \times 5$ two-padding $+ 5 \times 5$ one-padding $= 155$ possible label combinations. This shows a great variety

of conversation dynamics. Based on these n-grams, sequences of symbols, representing accommodation levels in a conversation, can be probabilistically generated. The predicted accommodation label l after two observed convergence labels c is taken from the probability distribution $p(l \mid c,\ c)$. Table 8.1 shows examples of such probabilistically generated sequences for the first eight accommodation labels of a conversation. The variability measure is defined as

$$P(sequence) = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{num(label_i)}{2}\right|, \tag{8.2.2}$$

where N is the number of labels in the sequence and $num(label)$ maps a label to a numeric value between -2 ($d+$) and 2 ($c+$). The overall probability of the sequence is calculated by

$$sequence\ probability = \prod_{i=3}^{N} p(label_i \mid label_{i-2}, label_{i-1}). \tag{8.2.3}$$

The perplexity measure is the average of all perplexity scores of the individual labels in the sequence. It is worth noticing that the third sequence in the table has lower variability than the second, although its overall probability is lower and its mean perplexity is higher. That means that, based on this model, higher variability is sometimes the more likely evolution of in interaction. On the other hand, since in most contexts, the probability of label n is relatively high, sequences with more no-change predictions are more likely to be generated, on average.

8.3 Clustering and incremental variational generation

To achieve the goal of the generating behaviors based on extracted core behaviors, the behavior description method from Section 8.1.3 and the n-gram approach from Section 8.2 need to be combined. On top of this, turn-level variations can be introduced using the GPes, as explained in Section 8.1. This will result in a variational probabilistic generation based on core behaviors detected in the dataset.

First, clusters of participant behaviors are sought to detect general similarities in behavioral patterns. Two clustering methods were utilized, namely k-means and hier-

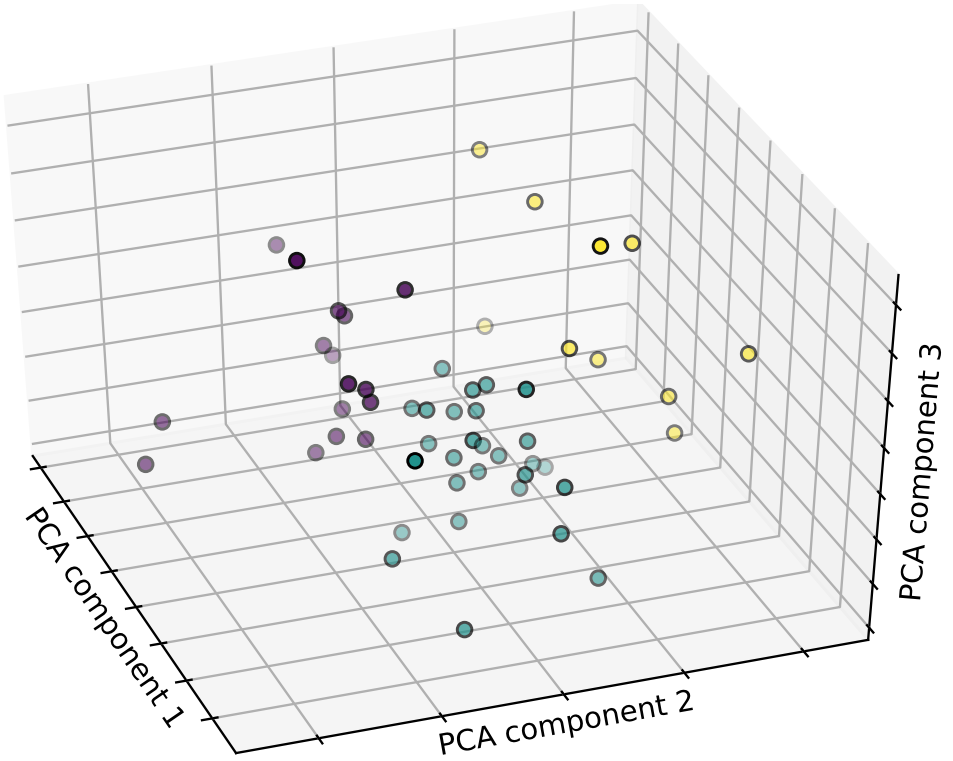


Figure 8.6: k-means clustering of the first three PCA components of the interactions' PAA sequences. Each circle represents one sequence in the three-dimensional space. A circle's color indicates the cluster to which the datapoint belongs.

archical linkage, each offering different insights on the data. The former method is a top-down approach with a pre-defined number of clusters, which offer a more general view on the patterns, and the latter is a bottom-up approach with no prior assumptions, where more individual differences can be observed. While it cannot be expected to find a completely distinct pattern for each speaker, some separable clusters and general tendencies are expected to emerge, both when inspecting individual speakers and the dataset as a whole. To determine the number of clusters k and the number of projection dimension d, the the clustering process was performed with two, three, and five clusters using the two and three first principal component analysis (PCA) components. The process was repeated 10 times to account for the algorithm's non-deterministic nature, with the combination of three clusters and three components best separating the data, on average. The resulted clusters are shown in Figure 8.6.

While top-down clustering uncovers general grouping of the interactions in the dataset,

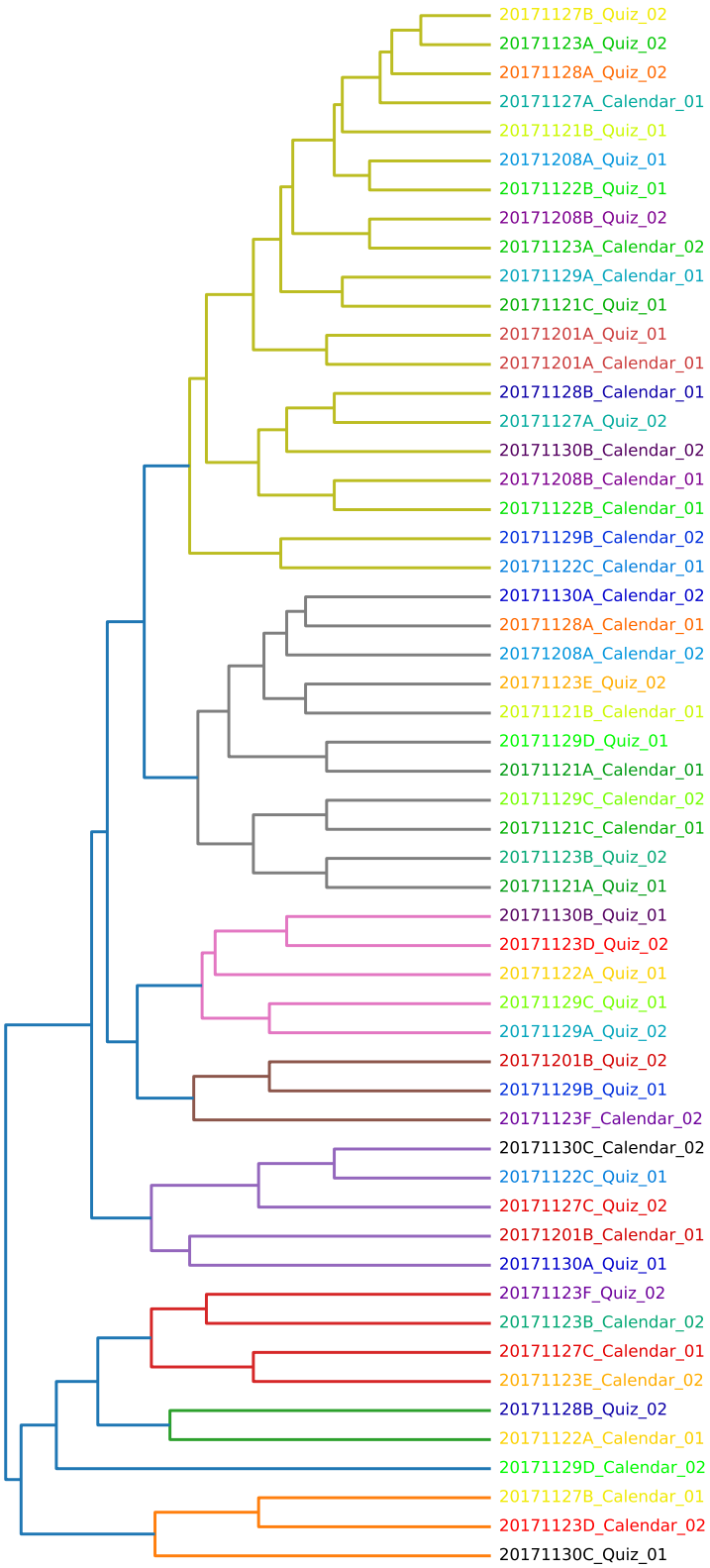


Figure 8.7: Dendrogram of the time series PAA representations of the interactions based on complete-linkage distances. Each cluster is represented by a different color of lines. Each leaf represent the interaction indicated by its label. Each participant's interactions pair is marked by labels of the same color. The leaf and cluster colors are *not* related. The leaves are order horizontally by their distance from left to right, so that the leaves of interactions that are more similar to each other (both inter- and within-cluster) are positioned closer. For example, the two interactions of participant 20171201A (12^{th} and 13^{th} leaves from the right) are the closet to each, as they are positioned together and within the same smaller sub-cluster. Contrarily, the labels of participant 20171128B are positioned the furthest from each other (6^{th} and 41^{st} leaves from the left).

bottom-up hierarchical clustering can reveal structural relations between them and measures their degrees of similarity. The distances between the 54 interactions can be measured and compared using their PAA values, as calculated in Section 8.2.1. The calculation was done using the *complete linkage* agglomeration technique (a.k.a. farthest point algorithm). This use of this linkage is motivated by the assumption that there are more general behavioral patterns to find beyond the speakers' individual behaviors, as it searches for the most dissimilar (and hence principally all) interactions in neighboring clusters and not only the closest one (single linkage). This method also allows using the entire PAA vectors without any pre-processing. Figure 8.7 shows the bottom-up distance clustering based on this linkage. Although, unsurprisingly, no definite order emerges, some general trends can be seen regarding the similarity between interactions of the same human speaker. Seeing that leaves are ordered horizontally based on their overall similarity, the distances between their positions can be utilized to determine each participant's behavior consistency. The average distance of the population is 16.5, far from the maximal possible average of 27, which points to a general tendency of speakers to behave similarly in their conversations. To reinforce this claim, dividing the space into three equal bins of "short", "medium", and "long" distances shows that 16, 10, and 1 interaction(s) fall into these bins, respectively. Moreover, this distribution has a median of 16 and its second tertiles located at 19, far from the "long distance" bin. Such skewness in the distribution indicates that speakers' behaviors are more often than not similar across interactions, regardless of any other factor (interaction length, task, order, and other factors discussed in Chapter 6). Notably, the two interactions of one participant, `20171201A`, even have the minimal possible distance of 1 between them (12^{th} and 13^{th} leaves from the right in Figure 8.7) and they are in the same lowest-level sub-cluster.

These clustering techniques show that different accommodative behaviors among speakers can be found and compared. Together with the probabilistic approach presented in Section 8.2.2, these can be used for generating sequences based on a specific behavior – or rather a group of latent behaviors. Ultimately, this would result in the system having a *core behavior* with *variations*, per the description in Section 2.1.1. To this end, an *incremental* generation process is needed. As opposed to the generation process demonstrated in Table 8.1, where the data for the entire interaction was known, an incremental generation is introduced here. An incremental generation better represents a live interaction, in which only the evidence accumulated up to a certain turn can be

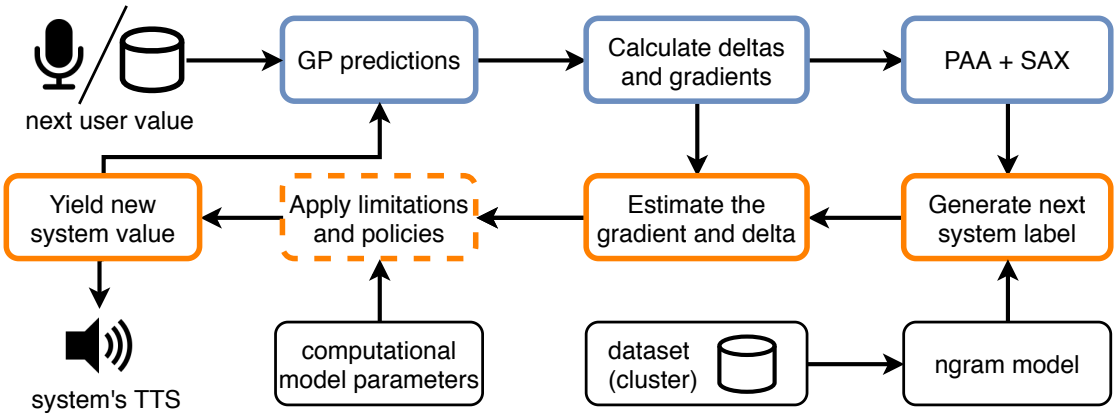


Figure 8.8: Schema of the incremental generation process. Blue boxes are related to the user phase and orange boxes to the system phase. Together, they complete a cycle of one round. The dashed orange frame marks an optional step. Note that the new value generated for the system is used both as output (e.g., for text-to-speech (TTS) synthesis) and as additional evidence for the next cycle. The user's values can be obtained either from online input or from some database.

used for analysis and prediction of the next turn. This can be utilized both for integration into a system and for simulating possible system behaviors for research. The process is summarized in Figure 8.8. It consists of the user and the system phases, which are roughly symmetric and with opposite goals: While the former assigns a label to an input value, the latter yields a value based on a label (which, in turn, is generated based on the label assigned to the user's input). A round is completed each time both phases were executed once. Hence, each round adds two values and their two corresponding labels, one for the user and one for the system. In the user phase, the latest system output and the new user input – obtained either from live input or a database – are added to the rest of the so-far accumulated evidence of their respective GPes, as done in Section 8.1.2. Then, deltas and gradients of these values are calculated (Figure 8.3) to subsequently create PAA and SAX sequences (Figure 8.5). The system phase starts with the SAX sequence contains one label per turn, including the user's new turn. The next system label is generated using the n-gram model with the probabilities acquired from the subset with the desired accommodative behavior (here, a cluster from Figure 8.6). This label represents a relative z-normalized change degree. Therefore, the value range covered by this label can be computed from all accumulated gradients, as indicated by the arrow between the second user step to the second system step in Figure 8.8. The system's GP

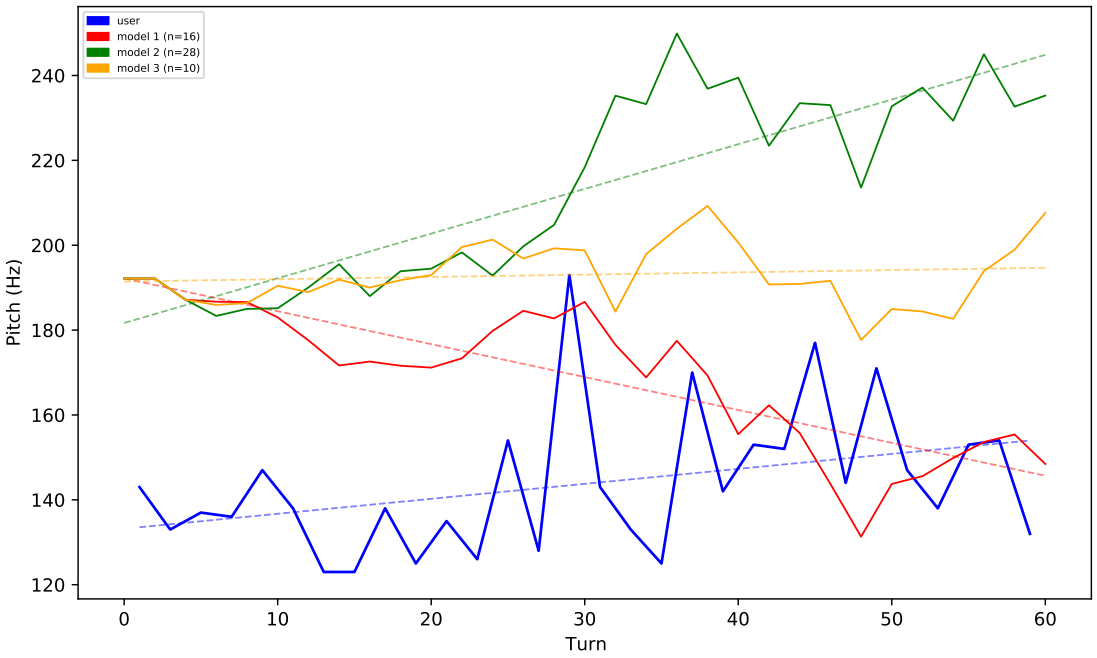


Figure 8.9: Responses of the three models, generated for the same user input (in blue). The dashed lines show the linear overall trend of the solid line of the same color. Note that these lines should be taken a reference for the general trend and **not as a representation or estimation of the values**.

from the first step is used to draw a single gradient value from that range. These two steps grant the variational property to the generation process; a label for the overall accommodation direction and a value for the specific amount of change. The drawn gradient value and the last system's gradient are used to determine the new gradient, from which the next system value can be calculated. The yielded value is then added to the accumulated evidence, so that it can be taken into account in the user phase's first step of the next round. In parallel, it can also be used as additional input for a text-to-speech (TTS) module of a spoken dialogue system, as demonstrated in Section 3.1 and Figure 9.1. External limitations or policies can intervene in this step to shape the system's output (see Section 7.2.5 for more motivation and examples). This combination of the statistical and computational models is further explored in Section 9.1.2.

This process was used to simulate accommodation changes during a conversation using the three clusters from Figure 8.6. Although the dataset used here is relatively small and is not designed to trigger different behaviors, the natural tendencies of the

participants should be, at least to some extent, reflected in this latent representation and be expressed in these simulations. For each cluster, an n-gram model was created as described in Section 8.2.2, based only on the interactions from this cluster. These models were then used to generate turn-level accommodation response sequences similarly to those in Table 8.1, but longer and with concrete values in addition to their labels. However, this time the generation was only for the system's side, given a pre-defined input of a human use, simulating a live interaction. In the example showed here, raw production values from interaction `20171201B_Calendar_02` were used. The first 30 rounds (60 turns) of three hypothetical conversations for this input were generated, one per model. N-grams of length ten were used, to take advantage of the larger context available in this longer interaction. All the models started with the mean value of the system productions in the dataset. This would be known also in a real-world scenario from the development of the system's TTS component and hence doesn't break the principle that only data available in a real-time conversation can be used. Figure 8.9 shows the simulated interactions. The trend lines show that, by and large, each model behaves differently: The green line tends towards divergence from the user, the red one inclines to convergence, and the yellow stays roughly the same regardless of the user's productions. These tendencies intensify once the user starts to show stronger variation around turns 35-40, and subsequently to a lesser extent till the end. Following these turns, the green line goes more sharply up, the red more sharply down, and the somewhat more neutral yellow starts to wiggle more. It can be claimed, therefore, that certain core behaviors were captured by the models, and they are sensitive to external input. This emphasizes the need to look at accommodation as a mutual process occurring in context over time and not as a discretized one-sided phenomenon, even if each speaker has an inert typical behavior. Noticeably, around turn 50 the red line goes down to fundamental frequency (f_0) values that might sound untypical for a female speaker (or just overly converging in general). This is due to the fact that the models here learn some theoretical probabilistic accommodative behavior, but do not have any knowledge about realistic human speech. Such issues can be easily addressed by applying policies based either on data and expert knowledge, as explained and demonstrated in Section 9.1.2.

IV

Application

Chapter 9

Accommodation Module for Spoken Dialogue Systems

PRODUCING accommodative speech output in spoken dialogue systems requires additional functionality for extracting speech properties from the user's input to consider when generating the system's output. This chapter introduces a module for SDS that adds such functionality to support vocal accommodation. The implementation and integration details as well as a demonstration of manipulating the system's output using this module are presented and discussed.

Table 9.1: Example of a feature definition used in the pipeline. The definition described the feature ə-*length*, i.e., length of a segment containing the phoneme [ə] (cf. Section 5.3.1.1).

Property	Value	Description
phoneme	AX	the phoneme that triggers this feature
context	.+ AX N	the context the phoneme must be in
initial	30	the starting value of this feature in the system
minimum	0	the smallest acceptable input value
maximum	80	the largest acceptable input value
measure	duration	the type of measure used for evaluation
pool size	5	the maximal number of exemplars in memory
update frequency	2	how often the feature's value is updated
calculation method	decaying average	the manner a new value is calculated
sensitivity	0.5	how quickly the feature changes
limit	0.8	how much the feature may change

9.1 Modularization

An implementation of the pipeline introduced in Section 7.2 is presented here as an independent module for spoken dialogue systems (SDSs). It is shown how the statistical components from Chapter 8 can be added to it to take advantage of the capabilities of both approaches. While supra-segmental features can be handled as well, only segmental features are discussed here, as they are less represented in accommodation modeling works and they require additional considerations, like segment context and local manipulation, as demonstrated in this chapter (as explained in Raveh and Steiner, 2017b). Thanks to the independence of this module, it can be added into any existing system that can provide the user's speech signal as input and a text-to-speech (TTS) module that can receive its output. The SDS presented in Chapter 10 has this module integrated into it to generate accommodative behaviors.

9.1.1 Accommodation pipeline

The implementations of the pipeline's steps will be explained using the feature definition with the properties described in Table 9.1. The entire implementation is described in Algorithm 2.

Algorithm 2: Phonetic responsiveness

Inputs : *ASRInput* – phonemes from recognized user speech
targetPhonemes – convergence features

Output: feature vectors with accommodated values

```
1  foreach (phoneme ∈ ASRInput) ∈ targetPhonemes do
2      feature ← phoneme.associatedFeature
3      context ← feature.phoneticContext
4
5      if not matches(phoneme, context) then          // filter based on context
6          break
7
8      if inRange(phoneme, feature.allowedRange) then // filter based on range
9          if poolSize=maxPoolSize then
10             deleteOldestExemplar()
11         feature.addExemplar(phoneme)
12     else
13         break
14
15     if toUpdate = 0 then
16         method ← feature.calculationMethod
17         poolValue ← method.calculate(pool)
18         newValue ← rate · poolValue + (1 − rate) · feature.currentValue
19         threshold ← convergenceLimit · poolValue
20
21         if newValue > threshold then               // limit accommodation
22             newValue ← threshold
23         feature.value ← newValue
24         toUpdate ← updatefrequency
25     else
26         toUpdate ← toUpdate - 1
27 end
```

The *ASRInput* (Line 1) must not only contain the n-best hypotheses, but also their corresponding phoneme sequences. For improving performance, using a single hypothesis is recommended for a small language model or when very short sentences are expected. Since only the target phonemes are considered by the pipeline (and suprasegmental features where the specific phonemes do not play a role), some ASR accuracy may be traded for better performance, depending on the application. For example, a CAPT system might rely solely on the realization of specific phonemes, regardless of what the user said or should have said, but the system's response must be quick.

9.1.1.1 Detecting exemplars

The first step in the pipeline is detecting segments in the user's utterance that can be ascribed to target features defined for the system. For that, an automatic speech recognition (ASR) engine that emits phoneme times is required. Here, CMU Sphinx[19] was used, with functionality to support the emission of phoneme-level information that was added for this purpose. The pipeline starts once a phoneme associated with a feature is detected. For example, the feature ə-*length* is triggered whenever the `phoneme label` *AX* is detected in the ASR stream, which stands for [ə] German CMU phonemeset. It is then evaluated using its defined `measure`; here, its duration. Other measures includes "formants" for vowel quality, "category" for categorical differences, and more. This step is performed for each feature separately against each recognized phoneme, as shown in Algorithm 2. Phonemes not associated with any feature are ignored.

9.1.1.2 Filtering exemplars

Seeing that segmental features are detected merely based on a phoneme in which they may occur, additional filtering is required to retain only those instances where the phenomenon they aim to capture indeed occurs. This filtering step comes to add any linguistic conditions relevant for the phonetic feature in question beside the phoneme itself. For example, the feature ə-*length* aims to capture the German phonological process of elision or epenthesis of ə in word-final <-*en*>. Therefore, only detected phonemes that occur in the relevant phonetic context (here, before a word-final n) should be considered. The regular expression defined in the `context` property (representing the schwa elision rule described in Equation 5.3.1) is matched against the surrounding of the detected phoneme. In addition a range of acceptable input values is defined by the properties `minimum` and `maximum`. This prevents unrealistic values from being considered as an instance of the feature, which might occur due to signal processing errors and inaccuracies of the ASR module or the measuring process of the feature's value. In the example of the ə-*length* feature, only values between 0 ms to 80 ms are allowed, as a segment cannot be shorter than 0 ms and schwa segments in this context are highly unlikely to be longer than 80 ms. This filtering verifies that all exemplars taken into account when calculating

[19] https://cmusphinx.github.io/

a new value for the feature (Section 9.1.1.4) are sensible and would be valid for a human listener, which prevents unstable behavior of the model.

9.1.1.3 Storing exemplars

The exemplars that remain after the filtering step are kept in the "memory" of the system to be used when a new value needs to be calculated (see Section 9.1.1.4). This memory is represented by a matrix, which contains vectors with the exemplars' values. Whenever a new exemplar is stored, its value vector is added to the memory matrix. The `pool size` property determines the size of the memory. When the memory reaches its maximal size, the oldest exemplar is "forgotten" to make space for the newest one. The memory functions like a queue of vectors in the form of

$$\mathcal{F}_{dimension-view} = \begin{bmatrix} v_{11} & v_{21} & \dots & v_{n1} \\ v_{12} & v_{22} & \dots & v_{n2} \\ \vdots & \vdots & \ddots & \vdots \\ v_{1m} & v_{2m} & \dots & v_{nm} \end{bmatrix}, \tag{9.1.2}$$

where each row refers to a single value of the feature, i.e., v_{nm} is the n-th value of the m-th exemplar.

9.1.1.4 Calculating a new value

The property `initial` determines the starting value of the feature in the system. This value may be outside the range allowed for new exemplars. The value of the property `update frequency` indicates the number of accumulated new exemplars between each update of the feature. In the example here, this value is 2, which means that an update is triggered after every 2 instances of the feature detected in the *detecting exemplars* step. A higher value means that updates will be less frequent and hence each exemplar will be taken into account in fewer updates. The new value of a feature is calculated using the exemplars stored in the *storing exemplars* step. This calculation is based on the `calculation method` set for the feature. Different calculation methods can represent different types of approaches to the way accommodation occurs in humans. For example, in the example of the ə-*length* feature, the decaying average simulates the assumption that people accommodate more to recent utterances by giving higher weights to newer

exemplars. Decaying average is defined here as

$$\mu_n = \frac{1}{N}\sum_{i=2}^{N}(\eta v_i + (1-\eta)\mu_{i-1}), \tag{9.1.3}$$

where N is the number of exemplars in memory, η is the decay rate, μ_{i-1} is the accumulated decaying average from the previous exemplar, and v_i is the value of the i-th exemplar. Any function g that maps a vector to a scalar (i.e., one feature value) and a function $\mathcal{G}$ which maps a matrix to a vector (the entire feature history) by applying g to all the rows in $\mathcal{F}$ can be used, as in Equation 9.1.4.

$$\mathcal{G} : \mathbb{Q}^{n\times m} \longrightarrow \mathbb{Q}^m, \qquad g : \mathbb{Q}^m \longrightarrow \mathbb{Q}. \tag{9.1.4}$$

This enables experimentation with different accommodation strategies. The property `sensitivity` is used to simulate different levels of human tendencies to accommodate. It determines the balance between the current value and the newly calculated value from the exemplar memory when calculating a new value for the feature. In the example here, the value 0.5 sets an equal weight for the existing value and the user's input. Lower values result in a slower accommodation process, while higher values lead to a faster (and potentially more abrupt) process. This balance is defined by

$$\Upsilon \equiv \mathcal{C}_u = \rho\upsilon + (1-\rho)\mathcal{C}_{u-1}, \tag{9.1.5}$$

where $\mathcal{C}_u$ is the new feature value, υ is the newly calculated feature value after applying $\mathcal{G}$, $\mathcal{C}_{u-1}$ is the current value of the feature, and ρ is the convergence rate. A ρ value of 0 means that the exemplars are ignored, i.e., no convergence occurs and the current value is retained; a value of 1 will result in complete convergence, i.e., the current value is ignored. While a typical `sensitivity` value would be between 0 and 1, smaller and greater values could be meaningful in some applications to achieve over-divergence or over-convergence, respectively. As shown in Part II, peoples' accommodation may vary considerably in human-human interactions (HHIs) and human-computer interactions (HCIs) and different settings. This parameter can help to tune the system's behavior to achieve the desired behavior in the interaction. For instance, in a tutoring system

for pronunciation training, the desired behavior might be for the system to diverge from users' input when it detects that their pronunciation is wrong. By reflecting the users' utterance with overly diverged pronunciation (instead of explicitly pointing out their mistake), the user receives auditory feedback in a more implicit and "conversational" learning process.

9.1.1.5 Setting the new value

The process ends with transferring the new feature value to the system's TTS component, so that it can be used the next time this feature appears in the system's output. After this step, this new value will be used whenever this feature is used by a system using this model (as shown in Section 9.2.1). This step is responsible for an important issue, namely preventing or allowing *user imitation*. After some turns, it might happen that the model would calculate a value very close to the user's input and then continues to follow the user's production values, resulting in imitation of the user. Depending on the feature, this might sound weird to the user or even be perceived as mocking. To avoid that, the new value is limited in how close it is allowed to get to the exemplars. This is regulated by the property `limit`, which defines the maximally allowed proximity (in percentage) to the user: Setting this property to 0.8, as in the example feature used here, means that the value from the previous step is limited to 80 % of the difference between the current value and the accumulated exemplars. A value of 1 allows the new value to be as similar as 100 % to the exemplars in memory, i.e., no limitation. This limit is defined as

$$\Lambda = \delta \upsilon (1 - \lambda), \tag{9.1.6}$$

where Λ is the maximum convergence value allowed, δ is set to 1 if the system's values are increasing comparing to the user or -1 in case they are decreasing, and λ is the value of the `limit` property. Note that the value of this limit depends on the *direction* in which the accommodation occurs (convergence or divergence), which is determined by

$$\delta = \begin{cases} 1 & \text{if } \upsilon \geq \mathcal{C}_t \\ -1 & \text{otherwise} \end{cases}. \tag{9.1.7}$$

That is, if the values need to increase in order to become more similar to the user, the

limit's value will be smaller than the memory value, as vice versa. Ultimately, the final updated value for the feature is determined as follows:

$$\Upsilon = \begin{cases} v - \Lambda \text{ (limited)} & \text{if } v - \Upsilon \leq \delta\Lambda \\ \Upsilon \text{ (unchanged)} & \text{otherwise} \end{cases}. \tag{9.1.8}$$

It is important to mention that this limitation is not artificially added here, but is was also observed in the data of the empirical experiments in Chapters 4 to 6, where humans typically accommodated only up to a certain limit.

9.1.2 Combining the computational and the statistical models

The pipeline implementation in Section 9.1.1 offers basis for accommodation capabilities based on the computational model presented in Chapter 7. The advantage of this realization is the ability to directly control the manner in which the accommodation occurs. This not only allows to craft and experiment with well-defined behaviors, but also to utilize expert knowledge or findings from empirical data. For example, external knowledge regarding the typical values of features and the places they occur can be applied in the filter step to exclude instances that do not match these criteria. Moreover, the calculations of the update and limit steps can be designed to approximate specific patterns observed in experimental data. This way, a system's behavior can be precise and well-defined, with the trade-off of being repetitive and predictive. While this alone already grants a *responsive* behavior, it lacks the *profiled* and *variational* behaviors described in Section 3.3.3. These drawbacks are addressed by the statistical model presented in Chapter 8, which offers more dynamic output based on a core behavior extracted from given data. The advantages of this data-driven approach is the simulation of human behaviors directly from data, which is more authentic and saves the time of fine-tuning their manual crafting. The risk is the usage of unfiltered raw data, which may result in problematic clustering and by extension unstable or noisy output.

To benefit from the merits of both models, they can be used together. The input of the statistical model is an input value from the user and its output is a value to use as output for the system. This corresponds to the roles of the store and update steps of the pipeline (although the store step could still be utilized nonetheless). Therefore, The statistical model can replace these steps and will be triggered only when an instance as

passed the filtering step, which would reduce the influence of noise in the data. Although the output of the statistical model should already account for the overall degree of accommodation, the limit step can be optionally kept to monitor that no unreasonable values are generated. This way, the pipeline structure is kept with all its advantages, but its final output is not limited to a specific deterministic behavior. Combining the computational and statistical models grants the system the benefits of both and gives experimenters more degrees of freedom for creating different scenarios and applications.

9.2 Integration

9.2.1 Extended spoken dialogue system architecture

The accommodation pipeline described in Section 9.1 can be integrated into a SDS with a standard architecture as an additional module, as shown in Figure 9.1. This module relies on input from the ASR module and its output is consumed by the TTS module, as defined in the pipeline. Therefore, it is inserted as a new, direct link between these two components, in addition to their connections to the natural language understanding (NLU) and from the natural language generation (NLG) modules, respectively. This additional speech processing (ASP) module can be used for any purpose that requires the speech signal and not only its transcription. Here, this information is leveraged for accommodation by the TTS module, but other speech analyses could be useful also for the dialogue manager (DM) or NLG modules, e.g., when matching the system's response to the user's mood based on voice characteristics (see Rothkrantz et al., 2004; Braun et al., 2016).

9.2.2 Speech manipulation

Although the ASP module is not responsible for the synthesis of the system's speech output, it is important to show that it provides relevant information for speech synthesis. This is especially important since the synthesis, along with the required changes for expressing accommodation, need to be applied *in real-time* and cannot be learned offline prior to an interaction. The examples given here focus on segmental features, because real-time manipulation techniques for them are less common than for supra-segmental features like fundamental frequency (f_0) and articulation rate (AR). For segmental features, modifications are applied to specific sounds occurring within short timespans, as

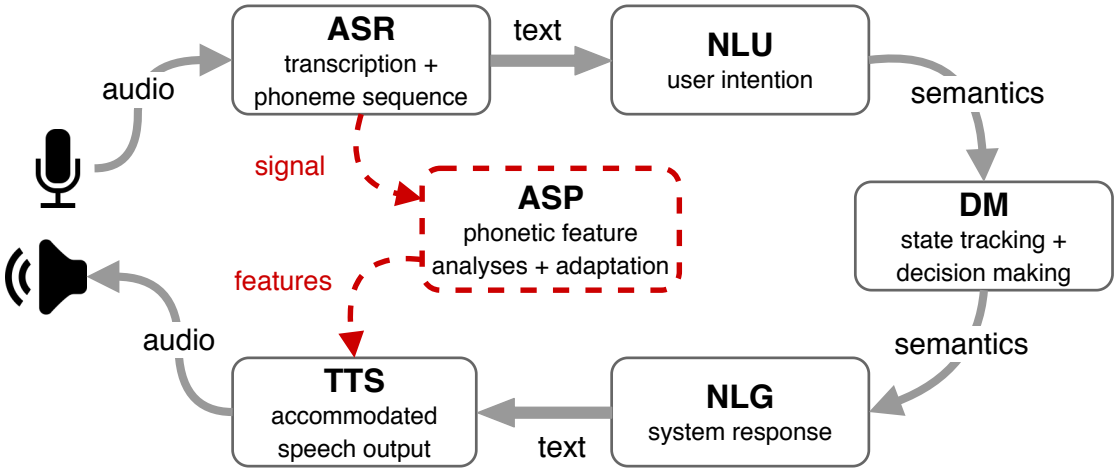


Figure 9.1: Suggested architecture for an accommodative spoken dialogue system (cf. base architecture in Figure 3.1). The added ASP module and its links from the ASR and to the TTS modules are colored in red.

oppose to supra-segmental features where longer segments and even an entire utterance are influenced. The challenge is, therefore, to apply these modifications with smooth transition from and to the surrounding segments and without creating artifacts.

The manipulation itself can be done either post-hoc on the audio signal itself or as part of the synthesis. The first approach relies on signal processing techniques that can be applied on the TTS output to achieve the desired modification. For this, the timespans of the target feature must be acquired from the TTS engine to detect the part in the signal that needs to be modified. Then, the appropriate process needs to be applied. For example, changing the format frequencies to manipulate vowel quality, shortening a segment to achieve [ə] elision, etc. Many of the manipulations can be done based on the source-filter theory (Fant, 1970), but any type manipulation would need to be written manually and run after the TTS finished generating the speech signal. The second approach relies on the TTS model to be able to capture the variations of the target feature. This might be difficult, especially for non-categorical features with relatively subtle changes. The model would need to be trained on a dataset that contains the feature's variations. As with any learning task, it is not guaranteed that the model will correctly learn them or be able to apply them correctly every time. On the other hand, if works property, this approach is more robust and should create fewer artifacts, because direct manipulation of the signal is avoided. However, direct manipulation might

be more accurate and offer direct control over the change in the signal which might not be achieved using a pre-trained model. These trade-off are an important consideration when choosing an approach, which might depend on the requirements of the application in question.

Both approaches were tested to realize accommodation changes. The first with the feature [ɛː] vs. [eː] which requires a manipulation in the vowel space continuum, and the second with the categorical feature [ç] *vs.* [k]. The source-filter manipulation was done as follows: First, the formant contours were extracted from the audio signal. this was done by computing the LPC coefficients with the algorithm by Burg, as given by Press et al. (1992). One value was extracted per formant every 6 ms with linear interpolation for missing values. Then, the first and second formants of the vowel target were changed separately using overlap-add (Hamon et al., 1989). The changes were based on the respective means of the formants in the duration of the target vowel, while taking the overall contour into account. Finally, The original signal was used as the source and was filtered by the manipulated formant contours, resulting in a new speech signal that differs from the original only by the target vowel's segment. The second approach was done with neural synthesis using Tacotron[20] (Shen et al., 2018), which is trained to generate spectral information directly from textual input, and the neural vocoder WaveGlow (Prenger et al., 2019). To capture the categorical allophonic variance, the system was trained on *phonemic* input (as opposed to usual grapheme-based representations), similarly to the training process in Fong et al. (2019) For this, the neutral voice subset of the PAVOQUE dataset[21] (Steiner et al., 2013) was used, which comprises ∼5.8 h of speech. The phonetic transcriptions were done automatically based on the transcriptions provided with the dataset and were manually verified and corrected. After training, the model was able to generate synthesized speech based on an input of an arbitrary phoneme sequence. This opens many possibilities, one of them is alternating between [ç] and [k] to express the variation of this feature. It is important to note that the other-category variant of a word was not included in any dictionary and was not seen by the model during training. Figure 9.2 shows an example of an original sentence (pronounced with [ç] sounds) and its manipulated form (with [k] sounds).

[20]Tacotron2 architecture based on the implementation on https://github.com/NVIDIA/tacotron2

[21]https://github.com/marytts/pavoque-data

Figure 9.2: Oscillograms and spectrograms of the words *"König"* (king) and *"wichtig"* (important) pronounced originally with [ç] (top), which were changed into [k] (bottom). The modified segments are marked with a star above the phoneme symbol in the transcriptions, and it can be seen that no artifacts were introduced in any other segment. The vertical lines show the phonemic segmentation (note that the segments are not perfectly aligned in the two productions). The x-axis shows the chronological phonetic transcription over time (2.34 s in total) and the spectrograms' y-axes show the frequencies up to 5,000 Hz.

Chapter 10

Web-Based Responsive Spoken Dialogue System

With the ability to simulate accommodation, a complete accommodative spoken dialogue system is introduced in this chapter. Its architecture and various customization possibilities are motivated and explained. The vocal changes are illustrated using dynamic visualizations in the system's graphical user interface. A replication of the human-computer interaction experiment is showcased to exhibit the system's capabilities and demonstrate its advantages.

10.1 Overview and key aspects

Simulating and triggering accommodation effects occurring in human-human interaction (HHI) in spoken dialogue systems (SDSs) takes them one step further toward human-like communication. The system presented in this chapter encapsulates the knowledge acquired from the experiments in Part II, the behavior designs developed in Part III, and the module introduced in Chapter 9. It contains mechanisms to track the states and changes of segment and suprasegmental phonetic features during a dialogue. All analyses are automated and run in real-time, which not only saves a lot of time and manual work typically needed in accommodation studies, but also makes the system more suitable for use in other applications. A user of the system may be a participant in an experiment, an experimenter designing an experiment or monitoring an ongoing experiment, or a researcher that uses it to analyze existing data. The system was designed with the following key principles in mind:

Focus on adaptation – the main goal of the system is to offer a tool for investigating vocal accommodation in human-computer interaction (HCI) for both online experiments and offline analyses. Putting vocal accommodation under the spotlight is the core novel contribution of the system, since very few systems offer such capabilities at all, and with control over the accommodative behavior in particular.

Customizability – the system includes several components that can be modified, either for changing the accommodation behavior itself (features, parameters, etc.) or for changing the settings (e.g., for different experiments). This allows experimentation with customized scenarios and configurations that can be easily compared in a controlled, reproducible environment.

Online scalability – the system can run in a web browser without any installations or additional files[22]. Since the system itself runs on a server, it is also possible to operate multiple instances, each with its own configurations and parameters. This makes it easy to distribute its use, e.g., for remotely conducting an experiment where a specific configurations can be given to each participant.

[22]Some features need to be enabled in the browser, like JavaScript and microphone access. However, any modern browser should not have any problem supporting all the necessary requirements. To increase performance, all speech analyses and processing are done on the server side.

This customizable system is a tool that could help to deepen the experimental possibilities and to automate the processes typically involved in accommodation-related experiments. The system's architecture, graphical user interface (GUI), and functionality are described in Sections 10.2 and 10.3. The experiment presented in Chapter 5 is replicated in Section 10.4 to demonstrate the system's utilization.

10.2 Architecture

As the system aims to offer a customizable playground for studying phonetic adaptation in HCI, a key property of its architecture is the separation between client-side, server-side, and external resources (see Figure 10.1). This separation makes it possible to run multiple clients on different machines at the same time with a single server collecting the data from all of them at the same time. The server, ideally running on a dedicated machine, is operated by a person responsible for designing and configuring the interactions, e.g., an experimenter. It collects information and audio recordings from all interactions with the system (which can be deleted afterwards for privacy purposes). This separation of the server grants the experimenter a lot of freedom and flexibility, since resources like feature configurations and dialogue domain can be modified independently, even while users are using the system. Additionally, multiple configurations can be prepared in advance (e.g., for different participant groups), regardless of the device the experiment will be performed on and without summoning the participants to the lab. Configurations can even be changed over the course of the experiment if additional variations are required. These configurations are transparent to the users, and no action is required from them aside from starting a new interaction with he system. This flexibility makes it easier and quicker to create new scenarios of interaction and to experiment with different features and parameters.

In addition to the technical advantages, letting users to interact with the system on a separate machine broadens the usage possibilities. For example, an experiment can be carried out remotely, without the need to invite participants to the recording studio one by one. Furthermore, as the connection to the server is done via a web browser, participants can connect use the system with their own computers wherever and whenever it suits them, without any additional installation or technical configurations. All of these make it possible to collect data from many users rapidly and easily. The main components of the system are the SDS with the accommodation module (Section 10.2.1),

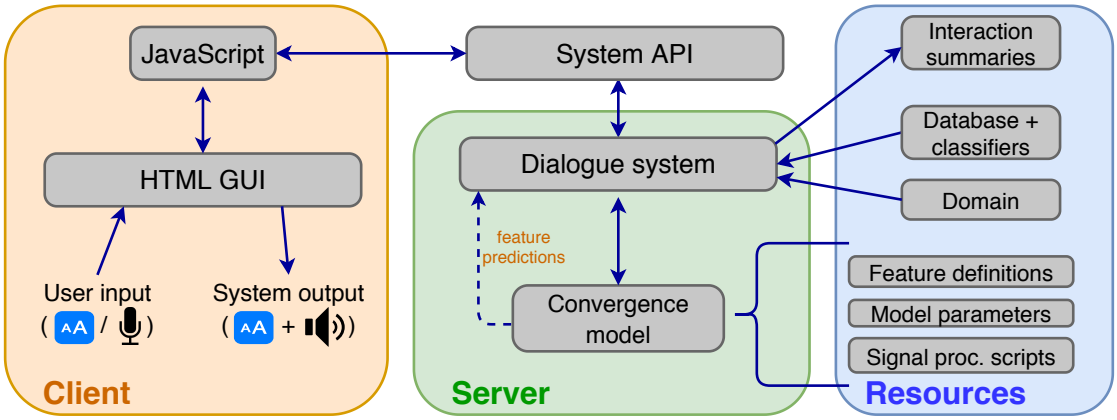


Figure 10.1: The architecture of the system. The background colors distinguish between client components, server components, and customizable external resources. The dashed line coming out of the convergence model's box indicates that the feature predictions may or may not be passed from the model to the system depending on the feature definition and update parameter (see Section 7.3).

the GUI (Section 10.2.2), and the external resources and configuration (Section 10.2.3).

10.2.1 Dialogue system

The core of the system is the dialogue system component (green block in Figure 10.1), which controls the flow of the interaction, processes users' inputs, and generates the system's responses. It uses the extended architecture presented in Section 9.2.1, which consists of traditional SDS components such as natural language understanding (NLU) and a dialogue manager (DM), but also contains the additional speech processing (ASP) module that adds accommodation support (Raveh and Steiner, 2017b). The implementation of this module in the system is as described in Figure 9.1. While the NLU component uses merely the transcription provided by the automatic speech recognition (ASR), the ASP module analyzes the speech signal itself. Concretely, it tracks occurrences of the defined features and passes their measured values to the convergence model, as explained in Section 10.2.3.1, which, in turn, forwards the tracked feature parameters to the text-to-speech (TTS) synthesis component. The TTS engine then takes the text generated by the natural language generation (NLG) component, and, if phonetic-level manipulation is supported by the TTS module, synthesizes the utterance using the values specified by the convergence model. The connection between the dialogue system's modules is managed by the OpenDial framework (Lison and Kennington, 2015, 2016).

The NLU and NLG modules are built using an OpenDial's domain file, as described in Section 10.2.3.2. Importantly, each of these components can be replaced with another implementation, all time it takes the same input and provides the same type of output.

10.2.2 Visualization and graphical user interface

The interaction with the system is done via an in-browser GUI (see screenshot example in Figure 10.2). At the top of the screen is a control bar, which offers the user an overview of the interaction and easy access to some general functionalities. On the left-hand side of the bar, the user can view the list of the interaction's turn history and jump to a specific one. It is also possible to see the list of tracked features and their current state. Both lists can be reset using the Reset button (in red), which starts a new interaction using the current configurations (which may have been changed during the ongoing interaction). On the other side of the bar, there are buttons for viewing on-screen how-to-use information window and changing various settings of the system, like convergence parameters, view options, resource location, etc. The rest of the GUI is divided into four areas: A *chat area* displaying the dialogue turns, an *interaction area* in which the user provides input to the systems, a *plot area* with interactive dynamic visualization of the tracked features, and a *notification area* where out-of-conversation messages for the user can be prompted. The functionality of these areas is described in Sections 10.2.2.1 to 10.2.2.4

10.2.2.1 Chat area

The interaction between the user and the system is shown in a chat-like format at the upper left part of the screen. Each turn's utterance appears inside a bubble with the user's and system's turns represented by different colors. A bubble always contains a single utterance, regardless of whether a floor change has taken place. A turn can be replayed at any time using the Play button next to the turn number, corresponding to the turn order on the list accessible from the control bar. Besides utterance bubbles, the system can also display general-purpose messages related to the interaction, which do not progress the dialogue flow and do count as system utterances. These messages can be used, for example, to give a participant additional information or further instructions during an experiment.

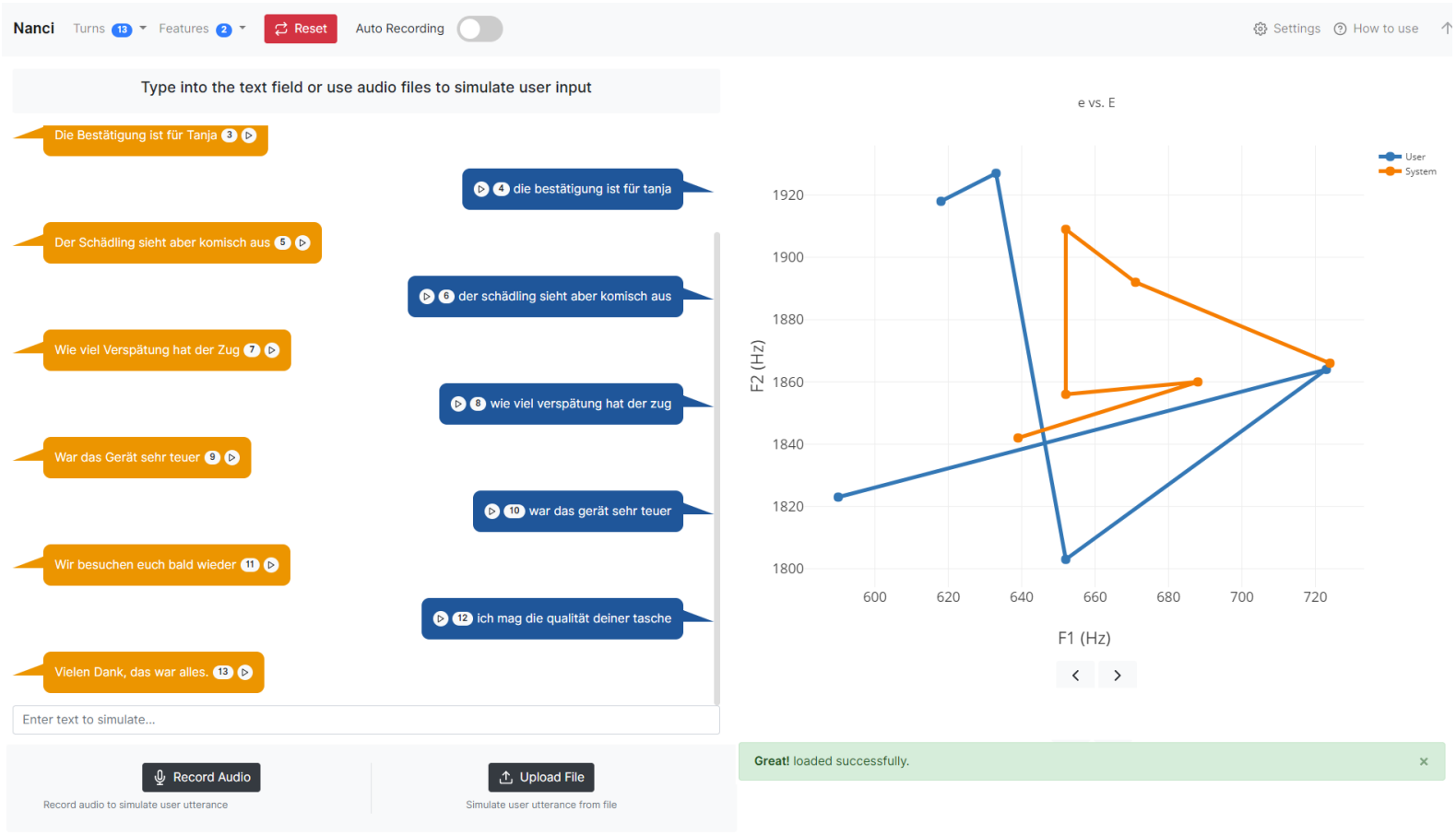


Figure 10.2: A screenshot of the system's graphical user interface, with the chat area on the top left, the interaction area on the bottom left, the plot area on the top right, and the notification area on the bottom right. In both the chat area and the plot area, the user and system are represented by the colors blue and orange, respectively.

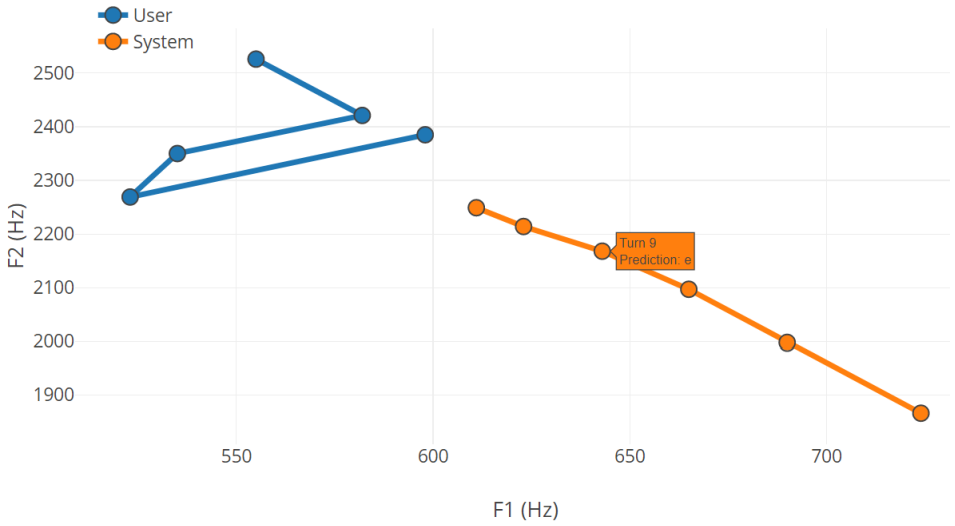


Figure 10.3: The plot area showing the states of the feature [ɛː] vs. [eː] during an interaction. The system's (orange, bottom right) gradually adapts to the user's (blue, upper left) detected realizations. A prediction of the feature's current realization is given for both interlocutors. The text box shows the mouse-over annotation of the turn in which the system's realization changed its vowel category.

10.2.2.2 Interaction area

The user can interact with the system with both written and spoken input using the controls at the bottom left of the screen. Spoken input can be provided either by speaking "live" into the microphone or via audio files with pre-recorded speech. These are typically useful for online and offline usage, respectively (as explained in Section 10.3), but pre-recorded utterances can also be useful for reproducing previous experiments or comparing different accommodation configurations with the exact same user input, as done in Appendix B. Text-based interactions progress through the dialogue (if applicable) and trigger any subsequent module, but will not affect the tracked features, as no vocal input is provided. This can be useful for quickly going through specific parts of an experiment (like instructions or setup) or for continuing the dialogue without changing the system's representation of the tracked features.

10.2.2.3 Plot area

Visualizations of the tracked features' changes over the course of the interaction are displayed in the upper right part of the screen. Each feature is visualized separately,

and new datapoints are dynamically added whenever applicable. Figure 10.3 shows an example of such a plot with several accumulated datapoints. The type of a feature's plot can be defined based on its characteristics, e.g., bars for one-dimensional features and lined scatter plots for two-dimensional features. These plots are generated using Plotly[23], which provides some interactive functionalities. Hovering over a datapoint in the plot reveals additional information, such as the turn in which it was added, or the realized variant of the feature produced in that turn, as predicted by its classifier (see Section 10.4.2).

10.2.2.4 Notification area

Whenever a message outside the content of the interaction needs to reach the user, it can be shown at the bottom right part of the screen. Such messages may include indications of the system's activity, e.g., successful initialization of the interaction, warnings and errors while uploading files, etc. The notifications can be colored blue, green, orange, or red to indicate the type of the message.

10.2.3 Customizations

The system aims to offer a platform for SDSs with convergence support that can be modified and customized according to the user's needs. All of the aforementioned system components can be customized, at least to some extent. This includes, among others, the phonetic convergence model, the features tracked by the system, and the dialogue domain.

10.2.3.1 Tracked features

The accommodation process is initiated by the phonetic features defined in the textual configuration file. The process is triggered whenever a phoneme associated with a segmental phonetic feature is detected in a segment by the ASR or for a suprasegmental feature that potentially occurs for in segment. The feature definitions may capture, for example, general tendencies or specific phonological rules, like schwa elision in German (see Equation 5.3.1). As explained in Section 9.1.1, each feature is detected and fil-

[23] https://plot.ly

tered based on its definition. This definition can be easily changed to experiment with different accommodation effects. An example for a feature definition is presented in Section 10.4.1.

10.2.3.2 Dialogue domain

The dialogue's flow is specified using OpenDial's XML-based format[24]. This format offers a structure for building models, rules, and conditions, which define the DM logic. The rules connect between intents provided by the NLU component to the output generated by the NLG component. Additional parameters are used to trigger processing for other modules of the SDS, like the ASP module in the system discussed here. More details about building a domain file can be found in Lison and Kennington (2016). The format of the domain file makes it easy to define new scenarios for the system, like different experimental settings. Rules are written mostly using regular expressions, which makes it relatively easy also for non-technical users to modify the system's logic. Since the DM keeps track of parameters from all modules, the system's output can even be influenced by the state of the accommodation state in the ASP module.

10.2.3.3 Speech processing

Multiple components of the system deal with different aspects of speech processing. As each module in the system can be replaced independently, different engines and models can be used. For example, the ASR engine can be replaced for improving performance or adding support for additional languages. The TTS component can be replaced as well, e.g., for changing the voice of the system or offering better control over phonetic manipulations. The tool used for the phonetic analysis can be changed as well to improve accuracy or performance. The models and tools described here are those that were used for the showcase presented in Section 10.4. The ASR component uses CMUSphinx[25] (Lamere et al., 2003), with an extension to the phoneme emission functionality to provide the ASP module the phonetic input information it needs (see Section 9.1.1). The acoustic

[24] http://www.opendial-toolkit.net/user-manual/dialogue-domains

[25] Sphinx4 version 5prealpha, https://cmusphinx.github.io/

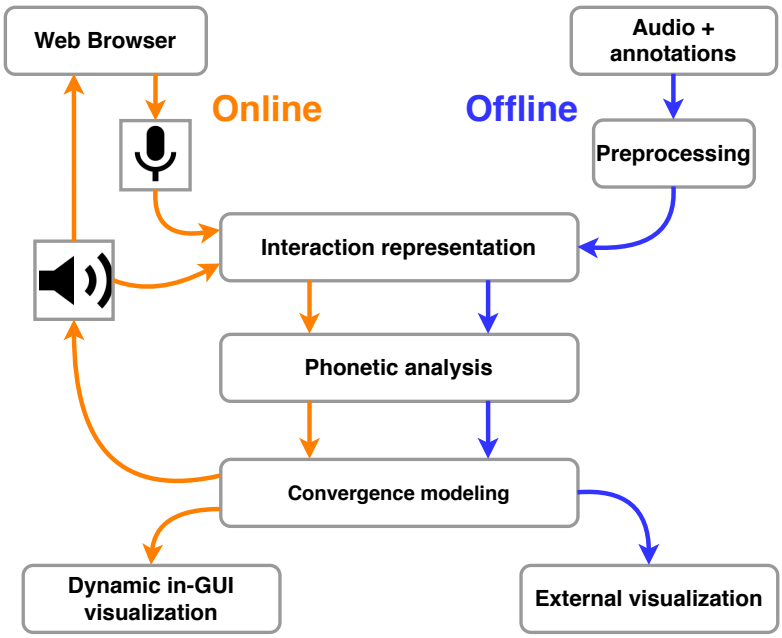


Figure 10.4: Online (orange) and offline (blue) modes of the system.

model and pronunciation dictionary were taken from CMUSphinx models[26]. A new language model was created especially for this purposes using SRILM (Stolcke, 2002). All the segmental and suprasegmental analyses required for the measuring accommodation were done using Praat (Boersma, 2018). MaryTTS (Schröder and Trouvain, 2003; Le Maguer and Steiner, 2017) was used as the TTS engine of the system, with `bits1-hsmm` and `bits3-hsmm` for its female and male voices, respectively.

10.3 Online and offline modes

The system can operate in two modes, as shown in Figure 10.4: Online – turn-based real-time interaction via the GUI (Section 10.2.2); and offline – on-demand analysis of existing interaction data, either from a recorded online session or a pre-recorded dataset. The accommodation-related parts of the system (roughly corresponding to the *server* and *resources* blocks in Figure 10.1) are common to the two modes, which makes it easy

[26]https://sourceforge.net/projects/cmusphinx/files/Acoustic%20and%20Language%20Models/German/

to switch between the two. The differences are in how the data is fed to the system and how the output is processed and visualized. The main difference is that in the online mode the process is recurrent and both the user's input and the system's current state are acquired on the fly. Depending on the scenario defined in the domain file, in online mode there could always be another turn, either of the user or the system, that will continue the interaction. In the offline mode, the data scope is, by definition, finite. Technically, the dataset is represented and processed the same way as online interactions, but there is no need to output the system's state to a user. The common representation also makes it easy to compare pre-recorded interactions with live ones. However, the output of the accommodation model after each turn is handled differently in both modes. In online interactions, the output is sent on a turn-by-turn basis to both the web-based GUI for visualization (as in Figure 10.3) and back to the user as auditory response from the system. The offline mode doesn't need to interact with a user, so the entire analysis is saved together, along with some additional analyses that can only be preformed with complete interactions. This output can then be used for further statistical analysis and be visualized separately using other tools (like those in Figures 2.1 and 6.7). For using the online mode, the system needs to run as a server and be accessed via a web browser. The offline mode can be used either directly from the command line or programmatically from another application. A mid-way usage is to manually load pre-recorded files via the GUI to simulate a live interaction, as explained in Section 10.2.2.2. While this requires more time and manual effort, it lets the user observe the visualized gradual phonetic changes.

10.4 Showcase: replicating a shadowing experiment

As a showcase of the system capabilities, it was utilized to replicate the shadowing experiment described in Section 5.3. The experiment is designed to trigger phonetic convergence by confronting the participants with stimuli in which certain phonetic features are realized in a way different from their natural production. This was done using the offline mode of the system, to simulate a real experiment and automate certain parts of it that would otherwise be performed manually. The replication used the original stimuli and utterances of the participants. However, analyses originally done post facto and to different extents manually, like detecting the realized variant, measuring the features' values, etc., are now done automatically. This demonstrates an

automated, reproducible execution, and also offers additional insights via classification of feature realizations and dynamic real-time visualizations. Finally, using the system the experiment becomes more of a fluent dialogue rather than a experimental simulated interaction, which enhances its HCI nature.

10.4.1 Setup

For the experiment replication, two of the three segmental features investigated in the original experiment were used. In addition to the ə-length feature shown in Section 9.1.1 the feature [ɛː] vs. [eː] was included. Both features are described in detail in Section 5.3.1.1, and Table 10.1 shows example stimulus sentences containing them. As in the original experiment, the word containing the target features were embedded into 15 short carrier sentences and 25 filler sentences, in which none of the features occur (see Appendix A for the full stimulus list). Although both features' underlying values are gradual, they are perceived as two-way categorical variations. To map these underlying values to a specific variant, a classifier was associated with each feature, as explained in Section 10.4.2. The definition of the [ɛː] vs. [eː] features was as follows:

```
- `e_E_vowel':
        phoneme: EHH
        context: '.* EHH .*'
        initial: 450 2100
        minimum: 300 1500
        maximum: 750 2900
        measure: formants
        calculation: decaying average
        sensitivity: 0.3
```

The values of the keys `minimum`, `maximum`, and `initial` stand for the first two formant frequencies. The `calculation` method for this feature is *decaying average* (Equation 9.1.3), which is similar to the regular average but with each value contributing exponentially less to the final value, so that the last (newest) exemplar contributes the most. Adding such property to the measure gives more weight to new exemplars that were received chronologically closer to the current turn and thus makes the change more strongly influenced by the productions closer to the accommodation change. Using this measure comes to support the analogy of the exemplar pool to short-term memory, which remembers recent events better than older ones.

Even though further aspects of the experiment could be automated, the experimental

War	das	Ger**ä**t	sehr	teuer?
Was	*the*	*device*	*very*	*expensive?*
Wir	besuch**en**	euch	bald	wieder.
We	*will visit*	*you*	*soon*	*again.*

Table 10.1: Examples of stimuli containing the target features. Each sentence contains only one feature. A list with all target and filler stimuli can be found in Appendix A.

procedure stayed as faithful as possible to the procedure of the original experiment. The domain file created for the showcase was designed to substitute the role of the experimenter in the shadowing phase (cf. Figure 5.4), i.e., mainly presenting and playing the stimuli to the participant. The stimulus order from the original experiment's baseline phase was preserved and semi-randomized in the shadowing phase using the same logic as in the original. It was also configured to perform the transitions between the phases. Although it should be assumed that the user indeed repeats the presented utterance, the system nonetheless verifies that the user's utterance matches the current stimulus using the customized language model described in Section 10.2.3.3 before presenting the next stimulus.

10.4.2 Classifiers training

As mentioned above, a classifier can be defined for each tracked feature to let the system determine to which realization category each encountered exemplar belongs. This automates the annotation otherwise done manually by the experimenter during or after the experiment. In the original shadowing experiment, this includes both the determination of the participants' preferred variation in the baseline phase and the annotation of the participants' realizations in the shadowing phase. To that end, the associated classifier provides real-time classifications for both the user's and the system's realizations of that feature. This not only saves time, but also helps to prevent inconsistencies that on-the-fly manual annotation might yield. With this information available, more meaningful insights can be gained into the variation dynamics over the course of the interaction. In other applications, like computer-assisted pronunciation training (CAPT), this information may be taken into account when deciding on the system's next turn. The classifiers for the replicated experiment were trained on datasets corresponding to the target features' ranges. The ə-length classifier trained to classify segments shorter and

sensitivity (0 to 1)	0.2	0.3	0.4	0.5
adoption (%)	79	86	75	69

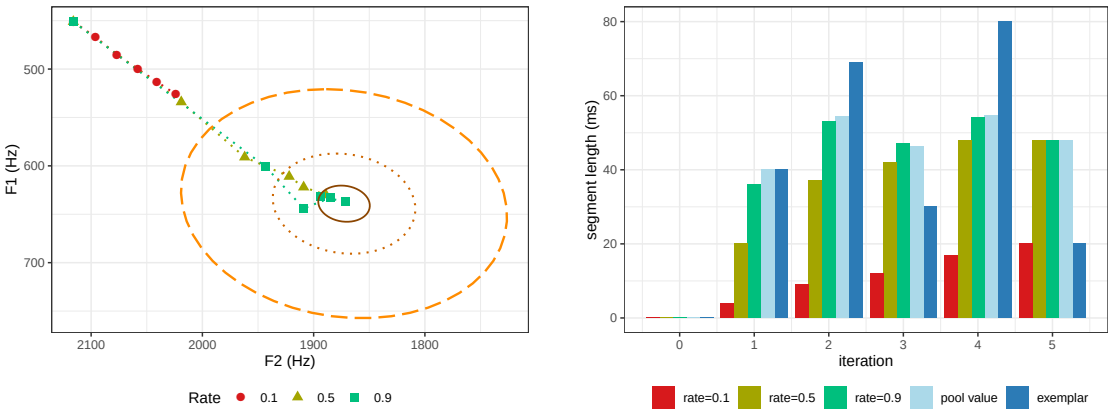
Table 10.2: The system's convergence degree with different degrees of sensitivity.

longer than 30 ms, and the [eː]/[ɛː] classifier was trained on F1 and F2 values of these vowels produced by female speakers (since for the replication a female participant was chosen as well as a female voice for the system). While training prior to the interactions is generally sufficient, online fine-tuning is also possible to update a feature's classifier whenever requested by the user, e.g., every time the accommodation model is updated.

Here, a sequential minimization optimization (SMO) (Platt, 1998; Platt, 1999) implementation of the support vector machine (SVM) classifier (Vapnik, 1998; Joachims, 2005) was used, as the two-way categories of these features are expected to be linearly separable. Each turn's predictions dynamically added as interactive annotations to the visualization of the relevant features, as illustrated in Figure 10.3. The training data used for each classifier contains only the productions of the corresponding target feature from a single stimulus set, since these are the productions to which the participants were exposed during the experiment. This provides relatively few – but at the same time very precise – datapoints for each classifier, which were obtained using the same signal processing technique as the data collected in the experiment. As explained in Section 5.3.1.3, multiple stimulus sets were used in the experiment. The classification of the system's production was performed based on the stimulus type the participant listened to. For instance, a classifier trained on the natural stimuli was used when the participant was listening to this stimulus set.

10.4.3 Validation

For the baseline phase, the degree to which the underlying convergence model accumulated enough data to adopt the user's variant of the feature was examined. Higher adoption rate indicates a more stable preferred variant of the participant. The participant's preferred variants was determined based on the majority vote at the end of this phase, as in the original experiment. For example, if the user realized one variant twice and another three times, the latter was considered the preferred one. Table 10.2 shows the adoption rates of the user's preferred variant as percentages of the mean preferred variant using different values of the *convergence rate* parameter (see Section 7.3). Inter-



(a) The effect of different convergence rates on the change of the system's representation toward the participant's *mean value* for the two-dimensional feature [eː] vs. [ɛː] using decaying average ($\mu = 0.3$). The points represent the calculation steps of the rates 0.1 (red circles), 0.5 (yellow tiranlges), and 0.9 (green rectangles). Each point is the new value calculated after encountering a new exemplar. The common starting point at the top left is the feature's defined initial value. The ellipses represent confidence levels of 90 %, 50 %, and 10 %.

(b) The effect of different convergence rates for the one-dimensional feature ə-length. The bars' height represent the feature's values on the y-axis with the calculation method set to simple average. The x-axis enumerates the feature's updates. The *exemplar* bar shows the feature's last added exemplar, and the *average* bar is the value of the feature after adding the last exemplar to the pool. Note that the 0-th update appears empty since the initial value of the feature was set to 0.

Figure 10.5: Illustration of the effect of different convergence rates on the updates of the system's realization of a two-dimensional feature (left) and a one-dimensional feature (right).

estingly, higher values do not necessarily result in higher percentages, due to systematic over-shooting the participant's production in each utterance. The value 0.3 provided the highest results and was therefore used through the rest of the replication. See Figure 10.5 for visualized examples of the convergence rate's influence. After obtaining the preference of each participant, the degree of convergence was examined per utterance in the shadowing phase. The participants were grouped based on their convergence behavior in the original experiment: One group of participants showed low to no tendency to converge (converged in ≤10 % of their utterances), the second had varying degrees of convergence (10 % to 90 %), and the third group of participants who were very sensitive to the stimuli (≥90 %). This grouping enables analyses based on collective vocal behaviors instead of individual differences. The groups were labeled *Low* (23 % of participants), *Mid* (50 %), and *High* (27 %), respectively. For validation purposes, the shadowing phase was treated as an annotation task of the realized variation in the par-

Table 10.3: Percentage of convergence cases and κ scores of the three-way convergence comparison for the three participant groups. Positive κ scores mean agreement between the annotations (here, the feature's realization category) and negative scores indicate disagreement. Scores close to zero point to an agreement occurring by chance.

	Convergence cases (%)			Cohen's Kappa (κ)		
	Sys-Stim	*Ref-Stim*	*Ref-Sys*	*Sys-Stim*	*Ref-Stim*	*Ref-Sys*
Low	<1	7	16	−0.57 ***	−0.08	0.17
Mid	22	23	32	−0.15 *	−0.15 *	0.27 ***
High	26	18	18	0.81 ***	−0.04	0.03
All	48	48	66	−0.11 *	−0.13 **	0.21 ***

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

ticipants' utterances, where a correct annotation (system produces same variation as the participant) indicates convergence. The three "annotators" are the stimuli themselves (*Stim*), the online classification of the system's representation of the feature (*Sys*), and labels from the training dataset used as references (*Ref*). The Cohen's kappa (κ) values[27] are shown in Table 10.3. Table 10.3 shows that *Ref-Sys* has $\kappa = 0.27$ (fair agreement) for the *Mid* group, but lower scores for the two other groups. This indicates that the reference values, which supposedly represent some universal average of the feature, indeed match the production of the participants that didn't deviate too greatly from their base production values, which reinforces the fact that the stimuli's influence on them was limited to either direction. The κ values for *Sys-Stim* describe how the system's representations matched the stimuli presented to the participants. Since the system accommodates to the participants' performance, these values exhibit how similarly the system's productions were to the productions of each participant group. The *High* group has $\kappa = 0.81$ (strong agreement), indicating a high similarity between these participants and the system, as expected. Contrarily, the κ value for the *Low* group is -0.57 (moderate negative agreement), showing that no convergence – and potentially even divergence – occurred with these participants. These results show that the online predictions made by the system presented here are capable of providing additional insights regarding the accommodation degrees occurring in an interaction.

[27] Calculated by the `kappa2` command of the `irr` R package, https://cran.r-project.org/package=irr

General Discussion

The gap between the measurable, objective methods to describe accommodation in experimental settings on the one hand and the free-form, subjective way it is used and perceived in everyday life on the other hand creates a great challenge in its research. Additionally, the large role individual differences play in both the production and reception of accommodation effects makes it hard to evaluate it, as there are no "correct" and "incorrect" labels that can be assigned to speakers' production. Moreover, the same production may lead to one effect with a certain interlocutor and a different effect with another – and both will be natural and acceptable. This has to do with many social and cognitive factors, including the speakers' personality and sensibility behavioral changes. Some people don't notice accommodation occurring in a conversation, including in their own speech, whereas some are sensitive even to subtle changes in their conversational partner's behavior. Their reactions may also be "holistic" or due to certain characteristics that do not necessarily correlate to properties observed directly in the acoustic signal (Babel and Bulatov, 2012). This makes even humans unreliable for evaluating accommodation, as opposed to many language processing tasks where human performance is set as a gold standard (or at least a goal to aim for). As a result, each study needs to introduce (or re-introduce) the used methods, which makes it hard to compare different studies and approaches. This lack of common measuring units and evaluation metrics is a major shortcoming of this research field. The overarching term "accommodation" holds various effects with different natures in it. Section 2.1.1 offers definitions for various terms based on their use in the literature. Having common terminology to describe them can be a step toward comparable common research methods and prevent confusions that stem from the use of different terms to describe the same effect or the same term to describe different effects.

An investigation of accommodation effects in pre-defined successful and failed conver-

sation was done in Chapter 4. Different effects were indeed found, especially with respect to the speaker who was leading the change. Finding that sales reps demonstrated more controlled and consistent triggering of convergence in the prospects' speech matches the assumption many in that business hold. However, the results of this experiment cannot be interpreted as causation and imply that they were more successful because of those convergence effects. Further experiments and comparisons with more reps are required for reaching this conclusion.

It is important to note that it was possible to reach these findings because of the emphasis on looking at accommodation as a dynamic phenomenon that unfolds over time. Without this temporal aspect, only more shallow broad conclusion can be drawn. Measuring accommodation as the difference between values at a few points (e.g., the beginning and end of an interaction) is an over-simplification of the process. First, the length of the interaction is not considered, which would lead to a similar conclusion for short and long interactions. Secondly, nuances in the mutual changes might be smoothed-out due to averages over long spans, as demonstrated in Figure 2.2 on page 26. A high temporal resolution grants a more fine-grained glance into patterns emerging in the data instead of in manually-picked datapoints. This approach is used, among others, in Section 6.4.2, in addition to distribution-based analyses, to obtain different points of view on the effects. The statistical model presented in Chapter 8 harnesses this idea by refraining from accurately defining behaviors and generating accommodative behaviors purely based on data, so that the temporal facets are implicitly included in it. Combining this data-driven mechanism with the cognitive-oriented approach from Chapter 7 adds human-centric motivation to the generated behavior. This fusion of computer-powered and human-motivated techniques has not been adequately explored so far, although it could offer a more comprehensive and explainable process.

To that end, one of the main goals of this thesis is to depict vocal accommodation as one comprehensive process that includes multiple related parts (see Figure 3.3), from examining effects in human-human interaction (HHI), via approaches to model them for computers, and of course the technical aspects of integrating and simulating them in spoken dialogue systems (SDSs). To get a better understanding of accommodation in human-computer interaction (HCI), it is important to see the connection between these parts and not investigate them in isolation from one another. More often than not, the technical side of spoken HCI (e.g., automatic speech recognition (ASR) accuracy

or text-to-speech (TTS) quality) are designed and developed separately from the user perspective. While this is understandable when aiming purely at performance improvements, it is problematic when addressing dialogue-related problems. However, since accommodation in HCI involves both computers and humans, both sides should be considered in the research and development of such systems. Ultimately, they need to work together to achieve better communication, just like human interlocutors in HHI. In the case of accommodation, investigating effects in humans that are not relevant or cannot be implemented in computers doesn't contribute to the advancement toward accommodative systems. Similarly, accomplishing technical goals that are not perceivable by humans or are not modeled in a human-centric fashion doesn't provide any added values as well. For instance, measuring convergence by the distance of mel-frequency cepstral coefficient (MFCC) vectors (as done by Han et al., 2018) might provide an interesting technical view on the matter, but since humans don't converge simply by sounding more alike, this is, doubtfully an efficient user-friendly way to implement accommodation in SDSs. Viewing accommodation as an involved interdisciplinary research topic encourages collaboration of researchers studying linguistics, humanities subjects like sociology and psychology, conversation and user-experience designers, engineers, and anything between them.

Systems with accommodative capabilities have been developed but showed varying degrees of fidelity. Although they are all described as accommodative systems (like the one introduced by Levitan et al., 2016), not all accommodation capabilities are born equal. This thesis distinguishes between several "levels of accommodation" in computers, as discussed in Section 3.3.3. Ranging from the mere ability to modify the system's speech ability to independently generating varying realizations of change, but also allows for customizable conversational design complexity depending on the target application. This concept is motivated by the parallelism to the assorted layers of accommodative behaviors in humans. For example, in normal, everyday conversation, people speak spontaneously, and therefore their speech will change freely based on their personality, personal preference, etc. This means that no specific behavior is consciously targeted here and the changes will be arbitrarily varied around this general behavior. In computers, this is paralleled to the variational generation around a "base" behavior of the system shown in Chapter 8, which, in turn, is extracted from different human productions. However, in other, more controlled situations, different accommodation strategies might be

more effective. Teachers use entrainment as a means for giving auditory feedback to language learners, by triggering an artificially strong effect to "draw" the student into a more correct articulation form, e.g., of a specific sound or intonation pattern. Although it is providing mostly implicit, this kind of feedback encourages learning from fluent, conversational responses. Still, this requires a different, more guided approach. Such an approach is realized in this work via the pipeline presented in Chapter 7, which offers deterministic control over the system's responsiveness using several cognition-oriented parameters. Such differentiation between accommodative mindsets has not been addressed before and it suggests more ways to model and implement accommodation in SDSs while keeping the specific goal and application in mind, e.g., chatbots with a free-form accommodation process in contrast to computer-assisted pronunciation training (CAPT) systems with a more well-defined goal. The system introduced in Chapter 10 offer a way to experiment with different configurations to achieve the desired behavior on the computer's side. This system can be extended, e.g., by developing more sophisticated accommodation models or supporting additional phonetic features These could be used, for instance, to replicate and automate more experiments (as done in Section 10.4) to accelerate and improve the data collection used for accommodation studies and offer better accommodation capabilities in computers.

Since computers are yet to possess full, human-level accommodation capabilities, it remains to be seen whether and how they will influence end-users once they do. First, like in the case of other human-inspired features like high-quality text generation and speech output, not all users might fancy such a capability that makes computers behave and perform more similarly to humans. One main reason for that the realistic yet imperfect attempt to adopt human behaviors often leads to the *uncanny valley* effect (Mori, 1970) and at some point makes users eerily uncomfortable (cf. Figure 1 in MacDorman, 2006). Secondly, as in HHI, some speakers are naturally less sensitive to phonetic changes and might not notice such variations in computers. While accommodation effects might still occur in that case, this raises the question of whether this would improve user experience nonetheless and whether developers would want to invest in features that users might not even acknowledge and appreciate. Finally, even when computers will have reached advanced accommodation capabilities (vocal and otherwise), they might not be accepted by users. Depending on the application and the agent type, people might not *want* their computers to demonstrate such human-like behaviors, especially if they don't necessarily

explicitly follow the user's preference. However, they might be useful and desirable in situations where the agent is designed to socially accompany a person for a long time. For instance, assistant social robots or therapeutic virtual humans that can realistic simulate HHI may achieve better rapport with their users, as the target is a closer long-term social relationship rather than the completion of isolated mundane tasks. Such tests will help reinforce HCI paradigms like Computers Are Social Actors (CASA). Somewhat ironically, this could only be thoroughly tested once such systems exist sometime in the future.

Bibliography

Abarbanel, Henry DI and Matthew B Kennel (1993). "Local False Nearest Neighbors and Dynamical Dimensions from Observed Chaotic Data". In: *Physical Review E* 47.5, p. 3057. DOI: 10.1103/PhysRevE.47.3057 (cit. on p. 64).

Abrego-Collier, Carissa, Julian Grove, Morgan Sonderegger, and CL Alan (Aug. 2011). "Effects of Speaker Evaluation on Phonetic Convergence". In: *International Congress of Phonetic Sciences (ICPhS)*. Hong Kong, pp. 192–195. URL: http://people.linguistics.mcgill.ca/~morgan/icphs2011Evaluation.pdf (cit. on p. 57).

Acosta, Jaime C and Nigel G Ward (2011). "Achieving Rrapport with Tturn-by-Turn, User-Responsive Emotional Coloring". In: *Speech Communication* 53.9-10, pp. 1137–1148. DOI: 10.1016/j.specom.2010.11.006 (cit. on p. 48).

Al Moubayed, Samer, Jonas Beskow, Gabriel Skantze, and Björn Granström (2012). "Furhat: A Back-Projected Human-Like Robot Head for Multiparty Human-Machine Interaction". In: *Cognitive Behavioural Systems*. Springer, pp. 114–130. DOI: 10.1007/978-3-642-34584-5_9 (cit. on p. 39).

Apostolico, Alberto, Mary Ellen Bock, and Stefano Lonardi (2003). "Monotony of Surprise and Large-Scale Quest for Unusual Words". In: *Journal of Computational Biology* 10.3-4, pp. 283–311. DOI: 10.1089/10665270360688020 (cit. on p. 148).

Aubanel, Vincent and Noël Nguyen (2020). "Speaking To a Common Tune: Between-Speaker Convergence in Voice Fundamental Frequency in A Joint Speech Production Task". In: *Plos one* 15.5. DOI: 10.1371/journal.pone.0232209 (cit. on p. 21).

Babel, Molly (2010). "Dialect Divergence and Convergence in New Zealand English". In: *Language in Society* 39, pp. 437–456. DOI: 10.1017/S0047404510000400 (cit. on p. 21).

Babel, Molly and Dasha Bulatov (2012). "The Role of Fundamental Frequency in Phonetic Accommodation". In: *Language and Speech* 55.2, pp. 231–248. DOI: 10.1177/0023830911417695 (cit. on pp. 4, 21, 78, 110, 189).

Babel, Molly, Grant McGuire, Sophia Walters, and Alice Nicholls (2014). "Novelty and Social Preference in Phonetic Cccommodation". In: *Laboratory Phonology* 5.1, pp. 123–150. DOI: 10.1515/lp-2014-0006 (cit. on pp. 21, 76).

Baddeley, Alan (2003). "Working Memory and Language: An Overview". In: *Journal of Communication Disorders* 36.3, pp. 189–208. DOI: 10.1016/S0021-9924(03)00019-4 (cit. on p. 137).

Bailly, Gérard and Amélie Martin (2014). "Assessing Objective Characterizations of Phonetic Convergence". In: *Interspeech*. Singapore, pp. 2011–2015. URL: https://www.isca-speech.org/archive/archive_papers/interspeech_2014/i14_2011.pdf (cit. on p. 22).

Baumann, Timo (2020). "How a Listener Influences the Speaker". In: *Speech Prosody*. Tokyo, Japan, pp. 970–974. DOI: 10.21437/SpeechProsody.2020-198 (cit. on p. 21).

Bell, Allan (1984). "Language Style as Audience Design". In: *Language in Society* 13.2, pp. 145–204. DOI: 10.1017/S004740450001037X (cit. on p. 23).

Bell, Linda, Joakim Gustafson, and Mattias Heldner (2003). "Prosodic Adaptation in Human-Computer Interaction". In: *International Congress of Phonetic Sciences (ICPhS)*. Barcelona, pp. 2453–2456. URL: https://www.internationalphoneticassociation.org/icphs-proceedings/ICPhS2003/p15_2453.html (cit. on pp. 4, 29).

Bellman, Richard (1957). "A Markovian Decision Process". In: *Journal of Mathematics and Mechanics*, pp. 679–684. URL: https://www.jstor.org/stable/24900506 (cit. on p. 147).

Benotti, Luciana, María Cecilia Martínez, and Fernando Schapachnik (2014). "Engaging High School Students Using Chatbots". In: *Innovation & technology in Computer Science Education*. ACM, pp. 63–68. URL: http://www.fundacionsadosky.org.ar/wp-content/uploads/2015/05/Engaging-High.pdf (cit. on p. 39).

Beňuš, Štefan (2014). "Social Aspects of Entrainment in Spoken Interaction". In: *Cognitive Computation* 6.4, pp. 802–813. DOI: 10.1007/s12559-014-9261-4 (cit. on pp. 6, 45).

Beňuš, Štefan, Marian Trnka, Eduard Kuric, Lukáš Marták, Agustín Gravano, Julia Hirschberg, and Rivka Levitan (2018). "Prosodic Entrainment and Trust in Human-

Computer Interaction". In: *International Conference on Speech Prosody*. Poznań, pp. 220–224. DOI: 10.21437/SpeechProsody.2018-45 (cit. on pp. 5, 29).

Benware, Wilbur A (1986). *Phonetics and Phonology of Modern German: An Introduction*. Georgetown University Press (cit. on p. 90).

Bergqvist, Amanda, Ramesh Manuvinakurike, Deepthi Karkada, and Maike Paetzel (2020). "Nontrivial Lexical Convergence in a Geography-Themed Game". In: *SIGdial*. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/2020.sigdial-1.26.pdf (cit. on pp. 19, 28).

Bernsen, Niels O., Hans Dybkjaer, and Laila Dybkjaer (1998). *Designing Interactive Speech Systems – From First Ideas to User Testing*. 1st ed. Springer-Verlag London. ISBN: 978-3-540-76048-1. DOI: 10.1007/978-1-4471-0897-9 (cit. on p. 48).

Black, Alan W (2003). "Unit Selection And Emotional Speech". In: *Interspeech*. URL: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.2.8457&rep=rep1&type=pdf (cit. on p. 92).

Black, Alan W and Paul Taylor (1997). *Festival Speech Synthesis System*. University of Edinburgh, Scotland, UK: Human Communication Research Centre (cit. on p. 36).

Boersma, Paul (2018). *Praat: Doing Phonetics by Computer*. URL: http://praat.org/ (cit. on pp. 59, 82, 95, 110, 182).

Bonferroni, Carlo E (1936). "Teoria Statistica Delle Classi e Calcolo Delle Probabilita'". In: DOI: 10.4135/9781412961288.n455 (cit. on p. 68).

Borrie, Stephanie A, Tyson S Barrett, Megan M Willi, and Visar Berisha (2019). "Syncing Up for a Good Conversation: A Clinically Meaningful Methodology for Capturing Conversational Entrainment in the Speech Domain". In: *Journal of Speech, Language, and Hearing Research*, pp. 1–14. DOI: 10.3886/E102720V2 (cit. on pp. 55, 60, 66, 67, 69).

Borrie, Stephanie A and Christine R Delfino (2017). "Conversational Entrainment of Vocal Fry in Young Adult Female American English Speakers". In: *Journal of Voice* 31.4, 513–e25. DOI: 10.1016/j.jvoice.2016.12.005 (cit. on p. 22).

Bourhis, Richard Y and Howard Giles (1977). "The Language of Intergroup Distinctiveness". In: *Language, Ethnicity, and Intergroup Relations*. Ed. by Howard Giles. London: Academic Press, pp. 119–135 (cit. on p. 21).

Branigan, Holly P, Martin J Pickering, Jamie Pearson, and Janet F McLean (2010). "Linguistic Alignment Between People and Computers". In: *Journal of Pragmatics* 42.9, pp. 2355–2368. DOI: 10.1016/j.pragma.2009.12.012 (cit. on p. 104).

Braun, Silke, Chiara Annovazzi, Cristina Botella, René Bridler, Elisabetta Camussi, Juan P Delfino, Christine Mohr, Ines Moragrega, Costanza Papagno, and Alberto Pisoni (2016). "Assessing Chronic Stress, Coping Skills, and Mood Disorders through Speech Analysis: A Self-Assessment "Voice App" for Laptops, Tablets, and Smartphones". In: *Psychopathology* 49.6, pp. 406–419. DOI: 10.1159/000450959 (cit. on p. 169).

Brennan, Susan E (1996). "Lexical Entrainment in Spontaneous Dialog". In: *ISSD* 96, pp. 41–44 (cit. on pp. 4, 15, 28).

Bulatov, Dasha (2009). "The Effect of Fundamental Frequency on Phonetic Convergence". In: *UC Berkeley PhonLab Annual Report* 5.5. URL: https://escholarship.org/uc/item/2w68f1c6 (cit. on p. 110).

Burkhardt, Felix and Walter F Sendlmeier (2000). "Verification of Acoustical Correlates of Emotional Speech Using Formant-synthesis". In: *Speech and Emotion*. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.385.471&rep=rep1&type=pdf (cit. on p. 92).

Carlson, Rolf, Jens Edlund, Mattias Heldner, A. Hjalmarsson, David House, and G. Skantze (2006). "Towards Human-like Behaviour in Spoken Dialog Systems". In: *Swedish Language Technology Conference (SLTC)*. Gothenburg, Sweden (cit. on p. 41).

Chartrand, Tanya L and John A Bargh (1999). "The Chameleon Effect: The Perception – Behavior Link and Social Interaction". In: *Journal of Personality and Social Psychology* 76.6, p. 893. DOI: 10.1037/0022-3514.76.6.893 (cit. on p. 18).

Cleveland, William S and Susan J Devlin (1988). "Locally Weighted Regression: An Approach to Regression Analysis by Local Fitting". In: *Journal of the American Statistical Association* 83.403, pp. 596–610. DOI: 10.1080/01621459.1988.10478639 (cit. on pp. 115, 118).

Coco, Moreno I and Rick Dale (2014). "Cross-Recurrence Quantification Analysis of Categorical and Continuous Time Series: An R Package". In: *Frontiers in Psychology* 5, p. 510. DOI: 10.3389/fpsyg.2014.00510 (cit. on pp. 63, 65).

Cohen Priva, Uriel, Lee Edelist, and Emily Gleason (2017). "Converging to the Baseline: Corpus Evidence for Convergence in Speech Rate to Interlocutor's Baseline". In: *The*

Journal of the Acoustical Society of America 141.5, pp. 2989–2996. DOI: 10.1121/1.4982199 (cit. on p. 22).

Cohen-Priva, Uriel and Chelsea Sanker (2019). "Limitations of Difference-in-Difference for Measuring Convergence". In: *Laboratory Phonology: Journal of the Association for Laboratory Phonology* 10.1. DOI: 10.5334/labphon.200 (cit. on p. 25).

Cohn, Michelle, Eran Raveh, Kristin Predeck, Iona Gessinger, Bernd Möbius, and Georgia Zellou (Oct. 2020). "Differences in Gradient Emotion Perception: Human vs. Alexa Voices". In: *Interspeech*. Shanghai, China. DOI: 10.21437/Interspeech.2020-1938 (cit. on p. 39).

Collins, Belinda (1998). "Convergence of Fundamental Frequencies in Conversation: If It Happens, Does It Matter?" In: *Spoken Language Processing*. Sydney, Australia (cit. on p. 78).

Coulston, Rachel, Sharon Oviatt, and Courtney Darves (2002). "Amplitude Convergence in Children's Conversational Speech with Animated Personas". In: *Interspeech*. Denver, CO, USA, pp. 2689–2692. URL: http://www.isca-speech.org/archive/icslp_2002/i02_2689.html (cit. on p. 29).

D'Imperio, Mariapaola, Rossana Cavone, and Caterina Petrone (2014). "Phonetic and Phonological Imitation of Intonation in Wwo Varieties of Italian". In: *Frontiers in Psychology* 5, p. 1226. DOI: 10.3389/fpsyg.2014.01226 (cit. on p. 78).

Davis, Katie, Joanna Christodoulou, Scott Seider, and Howard Earl Gardner (2011). "The Theory of Multiple Intelligences". In: *The Theory of Multiple Intelligences*. Ed. by RJ Sternberg and SB Kaufman, pp. 485–503. DOI: 10.1017/CBO9780511977244.025 (cit. on p. 56).

Day-O'Connell, Jeremy (2013). "Speech, Song, and The Minor Third: An Acoustic Study of the Stylized Interjection". In: *Music Perception: An Interdisciplinary Journal* 30.5, pp. 441–462. DOI: 10.1525/mp.2013.30.5.441 (cit. on p. 77).

De Jong, Nivja H and Ton Wempe (May 2009). "Praat Script to Detect Syllable Nuclei and Measure Speech Rate Automatically". In: *Behavior Research Methods* 41.2, pp. 385–390. DOI: 10.3758/BRM.41.2.385 (cit. on p. 110).

De Klerk, Dirk (1979). "Equal Temperament". In: *Acta Musicologica* 51.Fasc. 1, pp. 140–150. DOI: 10.2307/932181 (cit. on p. 82).

De Looze, Céline, Stefan Scherer, Brian Vaughan, and Nick Campbell (2014). "Investigating Automatic Measurements of Prosodic Accommodation and Its Dynamics in

Social Interaction". In: *Speech Communication* 58, pp. 11–34. DOI: 10.1016/j.specom.2013.10.002 (cit. on pp. 22, 25).

DeVault, David, Ron Artstein, Grace Benn, Teresa Dey, Ed Fast, Alesia Gainer, Kallirroi Georgila, Jon Gratch, Arno Hartholt, and Margaux Lhommet (May 2014). "SimSensei Kiosk: A Virtual Human Interviewer for Healthcare Decision Support". In: *International Conference on Autonomous Agents and Multi-Agent Systems*. Paris, France, pp. 1061–1068 (cit. on p. 40).

Dias, James W and Lawrence D Rosenblum (2016). "Visibility of Speech Articulation Enhances Auditory Phonetic Convergence". In: *Attention, Perception, & Psychophysics* 78.1, pp. 317–333. DOI: 10.3758/s13414-015-0982-6 (cit. on p. 76).

Dickey, Megan Rose (Dec. 2017). *The Echo Dot Was the Best-Selling Product on All of Amazon This Holiday Season*. URL: https://techcrunch.com/2017/12/26/the-echo-dot-was-the-best-selling-product-on-all-of-amazon-this-holiday-season/ (cit. on p. 37).

Duddington, Jonathan (2012). *eSpeak Text-to-Speech* (cit. on p. 36).

Duran, Nicholas D and Riccardo Fusaroli (2017). "Conversing with a Devil's Advocate: Interpersonal Coordination in Deception and Disagreement". In: *PloS one* 12.6, e0178140. DOI: 10.1371/journal.pone.0178140 (cit. on p. 60).

Dutoit, Thierry, Vincent Pagel, Nicolas Pierret, François Bataille, and Olivier Van der Vrecken (1996). "The MBROLA Project: Towards a Set of High Quality Speech Synthesizers Free of Use for Non-Commercial Purposes". In: *International Conference on Spoken Language Processing (ICSLP)*. Vol. 3, pp. 1393–1396. DOI: 10.1109/ICSLP.1996.607874 (cit. on pp. 92, 93).

Edlund, Jens, Mattias Heldner, and Joakim Gustafson (2006). "Two Faces of Spoken Dialogue Systems". In: *Workshop Dialogue on Dialogues: Multidisciplinary Evaluation of Advanced Speech-based Interactive Systems*. Pittsburgh, PA (cit. on p. 41).

Edlund, Jens, Mattias Heldner, and Julia Hirschberg (2009). "Pause and Gap Length in Face-to-Face Interaction". In: *Interspeech*. Heidelberg, Germany: Springer, pp. 2779–2782. ISBN: 978-3-642-12397-9. DOI: 10.1007/978-3-642-12397-9_13. URL: https://academiccommons.columbia.edu/doi/10.7916/D8H420RP/download (cit. on pp. 16, 22).

Edworthy, Judy, Elizabeth Hellier, Kathryn Walters, Wendy Clift-Mathews, and Mark Crowther (2003). "Acoustic, Semantic and Phonetic Influences in Spoken Warning

Signal Words". In: *Applied Cognitive Psychology: The Official Journal of the Society for Applied Research in Memory and Cognition* 17.8, pp. 915–933. DOI: 10.1002/acp.927 (cit. on p. 45).

Fant, Gunnar (1970). *Acoustic Theory of Speech Production*. 2. Walter de Gruyter. DOI: 10.1515/9783110873429 (cit. on p. 170).

Fernald, Anne (1991). "Prosody in Speech to Children: Prelinguistic and Linguistic Functions". In: *Annals of child development* 8, pp. 43–80 (cit. on p. 79).

Fong, Jason, Jason Taylor, Korin Richmond, and Simon King (2019). "A Comparison of Letters and Phones as Input to Sequence-to-Sequence Models for Speech Synthesis". In: *ISCA Speech Synthesis Workshop*, pp. 223–227. DOI: 10.21437/SSW.2019-40 (cit. on p. 171).

Foster, Mary Ellen, Andre Gaschler, Manuel Giuliani, Amy Isard, Maria Pateraki, and Ronald PA Petrick (2012). "Two People Walk into a Bar: Dynamic Multi-Party Social Interaction with a Robot Agent". In: *Multimodal Interaction*, pp. 3–10. DOI: 10.1145/2388676.2388680. URL: https://dl.acm.org/doi/pdf/10.1145/2388676.2388680 (cit. on p. 105).

Friedberg, Heather, Diane Litman, and Susannah BF Paletz (2012). "Lexical Entrainment and Success in Student Engineering Groups". In: *Spoken Language Technology Workshop (SLT)*. IEEE, pp. 404–409. DOI: 10.1109/SLT.2012.6424258 (cit. on p. 19).

Furnas, George W., Thomas K. Landauer, Louis M. Gomez, and Susan T. Dumais (1987). "The Vocabulary Problem in Human-System Communication". In: *Communications of the ACM* 30.11, pp. 964–971. DOI: 10.1145/32206.32212 (cit. on p. 16).

Gallois, Cindy and Howard Giles (2015). "Communication Accommodation Theory". In: *The International Encyclopedia of Language and Social Interaction*, pp. 1–18. DOI: 10.1002/9781118611463.wbielsi066 (cit. on pp. 4, 38, 40).

Galvez, Ramiro H., Lara Gauder, Jordi Luque, and Agustin Gravano (2020). "A Unifying Framework for Modeling Acoustic-Prosodic Entrainment: Definition and Evaluation on Two Large Corpora". In: *SIGdial*. Association for Computational Linguistics. URL: https://www.aclweb.org/anthology/2020.sigdial-1.27 (cit. on p. 143).

Gardner, Howard (June 1983). *Frames of Mind: The Theory of Multiple Intelligences*. Hachette Uk. DOI: 10.1086/414405 (cit. on p. 56).

Garrod, Simon and Martin J Pickering (2009). "Joint Action, Interactive Alignment, and Dialog". In: *Topics in Cognitive Science* 1.2, pp. 292–304. DOI: 10.1111/j.1756-8765.2009.01020.x (cit. on p. 18).

Gašić, Milica, Catherine Breslin, Matthew Henderson, Dongho Kim, Martin Szummer, Blaise Thomson, Pirros Tsiakoulis, and Steve Young (2013). "On-line Policy Optimisation of Bayesian Spoken Dialogue Systems Via Human Interaction". In: *Acoustics, Speech and Signal Processing (ICASSP)*. Vancouver, BC, Canada, pp. 8367–8371. DOI: 10.1109/ICASSP.2013.6639297 (cit. on p. 28).

Gijssels, Tom, Laura Staum Casasanto, Kyle Jasmin, Peter Hagoort, and Daniel Casasanto (2016). "Speech Accommodation Without Priming: The Case of Pitch". In: *Discourse Processes* 53.4, pp. 233–251. DOI: 10.1080/0163853X.2015.1023965 (cit. on pp. 40, 122).

Giles, Howard (1973). "Accent Mobility: A Model and Some Data". In: *Anthropological Linguistics* 15.2, pp. 87–105. URL: http://www.jstor.org/stable/30029508 (cit. on pp. 12, 21).

— (2007). "Communication Accommodation Theory". In: *The International Encyclopedia of Communication Theory and Philosophy*, pp. 1–7. DOI: 10.1002/9781118766804.wbiect056 (cit. on pp. 4, 12).

Giles, Howard, Nikolas Coupland, and Justine Coupland (1991). "Accommodation Theory: Communication, Context, and Consequence". In: *Contexts of Accommodation: Developments in Applied Sociolinguistics*. Ed. by Howard Giles, Justine Coupland, and Nikolas Coupland. Cambridge University Press, pp. 1–68. DOI: 10.1017/CBO9780511663673.001 (cit. on pp. 12, 21).

Gladwell, Malcolm (2006). *The Tipping Point: How Little Things Can Make a Big Difference*. Little, Brown. DOI: 10.1177/107769580105500412 (cit. on p. 54).

Glaser, Judith E (2016). *Conversational Intelligence: How Great Leaders Build Trust and Get Extraordinary Results*. Routledge. ISBN: 978-1629561431 (cit. on pp. 56, 57).

Glaser, Judith E and Ross Tartell (2014). "Conversational Intelligence at Work". In: *OD Practitioner* 46.3, pp. 62–67. URL: https://organizationalperformancegroup.com/wp-content/uploads/2016/10/ODP-Conversational-Intelligence.pdf (cit. on p. 57).

Gnewuch, Ulrich, Stefan Morana, and Alexander Maedche (2017). "Towards Designing Cooperative and Social Conversational Agents for Customer Service". In: *Conference: International Conference on Information Systems (ICIS)* (cit. on p. 56).

Godfrey, J. J., E. C. Holliman, and J. McDaniel (Mar. 1992). "SwitchBoard: Telephone Speech Corpus for Research and Development". In: *Acoustics, Speech, and Signal Processing*. Vol. 1. Los Alamitos, CA, USA: IEEE Computer Society, pp. 517–520. DOI: 10.1109/ICASSP.1992.225858 (cit. on p. 20).

Goldinger, Stephen D. (1998). "Echoes of Echoes? An Episodic Theory of Lexical Access". In: *Psychological Review* 105.2, p. 251. DOI: 10.1037/0033-295x.105.2.251 (cit. on p. 76).

Gordon, Carla, Kallirroi Georgila, Hyungtak Choi, Jill Boberg, and David Traum (Nov. 2018). "Evaluating Subjective Feedback for Internet of Things Dialogues". In: *Workshop on the Semantics and Pragmatics of Dialogue (SemDial)*. Aix-en-Provence, France. URL: http://semdial.org/anthology/Z18-Gordon_semdial_0010.pdf (cit. on p. 42).

Gregory, S., S. Webster, and G. Huang (1993). "Voice Pitch And Amplitude Convergence as a Metric of Quality In Dyadic Interviews". In: *Language & Communication* 13.3, pp. 195–217. DOI: 10.1016/0271-5309(93)90026-J (cit. on p. 110).

Gueguen, Nicolas, Celine Jacob, and Angelique Martin (2009). "Mimicry in Social Interaction: Its Effect on Human Judgment and Behavior". In: *European Journal of Social Sciences* 8.2, pp. 253–259. DOI: 10.1007/978-1-4419-1428-6_1798 (cit. on pp. 18, 19).

Hall, T Alan (2011). *Phonologie: Eine Einführung*. Walter de Gruyter. DOI: 10.1515/9783110215885 (cit. on p. 17).

Hamon, Christian, E Mouline, and Francis Charpentier (1989). "A Diphone Synthesis System Based on Time-domain Prosodic Modifications of Speech". In: *Acoustics, Speech, and Signal Processing*. IEEE, pp. 238–241. DOI: 10.1109/ICASSP.1989.266409 (cit. on p. 171).

Han, Jing, Maximilian Schmitt, and Björn W. Schuller (Sept. 2018). "You Sound Like Your Counterpart: Interpersonal Speech Analysis". In: *Speech and Computer SpeCom*. Ed. by Alexey Karpov, Oliver Jokisch, and Rodmonga Potapova. Vol. 11096. Lecture Notes in Computer Science. Leipzig, Germany: Springer, pp. 188–197. DOI: 10.1007/978-3-319-99579-3_20 (cit. on p. 191).

Harrington, Jonathan (2007). "Evidence for a Relationship Between Synchronic Variability and Diachronic Change in the Queen's Annual Christmas Broadcasts". In:

Laboratory Phonology 9, pp. 125–143. URL: https://www.phonetik.uni-muenchen.de/~jmh/research/papers/Harrington_proofs.pdf (cit. on p. 23).

Harrington, Jonathan, Sallyanne Palethorpe, and Catherine Watson (2000a). "Monophthongal Vowel Changes in Received Pronunciation: An Acoustic Analysis of The Queen's Christmas Broadcasts". In: *Journal of the International Phonetic Association* 30.1-2, pp. 63–78. DOI: 10.1017/S0025100300006666 (cit. on p. 23).

Harrington, Jonathan, Sallyanne Palethorpe, and Catherine I Watson (2000b). "Does the Queen Speak the Queen's English?" In: *Nature* 408.6815, pp. 927–928. DOI: 10.1038/35050160 (cit. on p. 23).

Heldner, Mattias, Jens Edlund, and Julia Bell Hirschberg (Aug. 2010). "Pitch Similarity in the Vicinity of Backchannels". In: pp. 1–4. DOI: https://doi.org/10.7916/D8WS92R4 (cit. on p. 25).

Hunt, Andrew J and Alan W Black (May 1996). "Unit Selection in a Concatenative Speech Synthesis System Using a Large Speech Database". In: *Acoustics, Speech, and Signal Processing*. Vol. 1. IEEE, pp. 373–376. DOI: 10.1109/ICASSP.1996.541110 (cit. on p. 92).

Huttenlocher, Janellen, Marina Vasilyeva, and Priya Shimpi (2004). "Syntactic Priming in Young Children". In: *Journal of Memory and Language* 50.2, pp. 182–195. DOI: 10.1016/j.jml.2003.09.003 (cit. on p. 17).

Hwang, Jiwon, Susan E Brennan, and Marie K Huffman (2015). "Phonetic Adaptation in Non-Native Spoken Dialogue: Effects of Priming and Audience Design". In: *Journal of Memory and Language* 81, pp. 72–90. DOI: 10.1016/j.jml.2015.01.001 (cit. on p. 16).

Ibrahim, Omnia, Gabriel Skantze, Sabine Stoll, and Volker Dellwo (2019). "Fundamental Frequency Accommodation in Multi-Party Human-Robot Game Interactions: The Effect of Winning or Losing". In: *Interactions* 20, p. 21. DOI: 10.21437/Interspeech.2019-2496 (cit. on pp. 25, 39, 105).

Iio, Takamasa, Yuichiro Yoshikawa, Mariko Chiba, Taichi Asami, Yoshinori Isoda, and Hiroshi Ishiguro (2020). "Twin-Robot Dialogue System with Robustness Against Speech Recognition Failure in Human-Robot Dialogue with Elderly People". In: *Applied Sciences* 10.4, p. 1522. DOI: 10.3390/app10041522 (cit. on p. 39).

Jackendoff, Ray (2009). “Parallels and Nonparallels Between Language and Music”. In: *Music Perception: An Interdisciplinary Journal* 26.3, pp. 195–204. DOI: 10.1525/mp.2009.26.3.195 (cit. on p. 77).

Joachims, Thorsten (2005). “A Support Vector Method for Multivariate Performance Measures”. In: *International Conference on Machine Learning (ICML)*. Bonn, Germany, pp. 377–384. DOI: 10.1145/1102351.1102399 (cit. on p. 186).

Jonell, Patrik, Mattias Bystedt, Per Fallgren, Dimosthenis Kontogiorgos, José Lopes, Zofia Malisz, Samuel Mascarenhas, Catharine Oertel, Eran Raveh, and Todd Shore (May 2018). “FARMI: A FrAmework for Recording Multi-Modal Interactions”. In: *Language Resources and Evaluation Conference (LREC)*. Miyazaki, Japan. URL: http://www.lrec-conf.org/proceedings/lrec2018/pdf/713.pdf (cit. on p. 105).

Jucks, Regina, Bettina-Maria Becker, and Rainer Bromme (2008). “Lexical Entrainment in Written Discourse: Is Eexperts’ Word Use Adapted to the Addressee?” In: *Discourse Processes* 45.6, pp. 497–518. DOI: 10.1080/01638530802356547 (cit. on p. 19).

Kang, Yoonjung (2010). “The Emergence of Phonological Adaptation from Phonetic Adaptation: English Loanwords in Korean”. In: *Phonology* 27.2, pp. 225–253. DOI: 10.1017/S0952675710000114 (cit. on p. 16).

Kawahara, Hideki (2006). “STRAIGHT, Exploitation of the Other Aspect of VOCODER: Perceptually Isomorphic Decomposition of Speech Sounds”. In: *Acoustical Science and Technology* 27.6, pp. 349–353. DOI: 10.1250/ast.27.349 (cit. on p. 93).

Kennel, Matthew B, Reggie Brown, and Henry DI Abarbanel (1992). “Determining Embedding Dimension for Phase-Space Reconstruction Using a Geometrical Construction”. In: *Physical review A* 45.6, p. 3403. DOI: 10.1103/PhysRevA.45.3403 (cit. on p. 64).

Keogh, Eamonn, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra (2001). “Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases”. In: *Knowledge and Information Systems* 3.3, pp. 263–286. DOI: 10.1007/PL00011669 (cit. on pp. 147, 148).

Kerly, Alice, Phil Hall, and Susan Bull (2007). “Bringing Chatbots into Education: Towards Natural Language Negotiation of Open Learner Models”. In: *Knowledge-Based Systems* 20.2, pp. 177–185. URL: http://www.ai-research.org.uk/AliceKerly/html/Kerly_Hall_AI-06_for_web.pdf (cit. on p. 39).

Kim, Midam, William S Horton, and Ann R Bradlow (2011). "Phonetic Convergence in Spontaneous Conversations as a Function of Interlocutor Language Distance". In: *Laboratory Phonology* 2.1, pp. 125–156. DOI: 10.1515/labphon.2011.004 (cit. on p. 21).

Kinsbourne, Marcel and J Scott Jordan (2009). "Embodied Anticipation: A Neurodevelopmental Interpretation". In: *Discourse Processes* 46.2-3, pp. 103–126. DOI: 10.1080/01638530902728942 (cit. on p. 4).

Kleinberg, Sara (Jan. 2018). *5 Ways Voice Assistance Is Shaping Consumer Behavior*. URL: https://www.thinkwithgoogle.com/_qs/documents/5604/1178-CES-Voice-Research-PDF.pdf (cit. on p. 38).

Kousidis, Spyros and David Dorran (Jan. 2009). "Monitoring Convergence of Temporal Features in Spontaneous Dialogue Speech". In: *Workshop on Speech Technology, Dublin, Ireland*. Dublin, Ireland. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.917.1366&rep=rep1&type=pdf (cit. on p. 143).

Kousidis, Spyros, David Dorran, Yi Wang, Brian Vaughan, Charlie Cullen, Dermot Campbell, Ciaran McDonnell, and Eugene Coyle (Sept. 2008). "Towards Measuring Continuous Acoustic Feature Convergence in Unconstrained Spoken Dialogues". In: *Interspeech*. Brisbane, Australia, pp. 1692–1695. URL: https://www.isca-speech.org/archive/archive_papers/interspeech_2008/i08_1692.pdf (cit. on p. 143).

Kovac, Mark and Jonathan Frick (2017). "It's 10 AM. Do You Know What Your Sales Reps Are Doing?" In: *Harvard Business Review*. URL: https://hbr.org/2017/03/its-10-am-do-you-know-what-your-sales-reps-are-doing (cit. on p. 54).

Kowatsch, Tobias, Marcia Nißen, Chen-Hsuan Iris Shih, Dominik Rüegger, Dirk Volland, Andreas Filler, Florian Künzler, Filipe Barata, Severin Haug, and Dirk Büchter (2017). "Text-Based Healthcare Chatbots Supporting Patient and Health Professional Teams: Preliminary Results of a Randomized Controlled Trial on Childhood Obesity". In: *Persuasive Embodied Agents for Behavior Change (PEACH)*. DOI: 10.3929/ethz-b-000218776 (cit. on p. 39).

Krivokapic, Jelena (2013). "Rhythm and Convergence Between Speakers of American and Indian English". In: *Laboratory Phonology* 4.1, pp. 39–65. DOI: 10.1515/lp-2013-0003 (cit. on p. 78).

Kuhl, Patricia K (1991). "Human Adults and Human Infants Show a "Perceptual Magnet Effect" for the Prototypes of Speech Categories, Monkeys Do Not". In: *Attention,*

Perception, & Psychophysics 50.2, pp. 93–107. DOI: 10.3758/bf03212211 (cit. on p. 89).

— (2004). "Early Language Acquisition: Cracking the Speech Code". In: *Nature Reviews Neuroscience* 5.11, pp. 831–843. DOI: 10.1038/nrn1533 (cit. on p. 89).

Lamere, Paul, Philip Kwok, Evandro Gouvea, Bhiksha Raj, Rita Singh, William Walker, Manfred Warmuth, and Peter Wolf (2003). "The CMU SPHINX-4 Speech Recognition System". In: *Acoustics, Speech and Signal Processing (ICASSP)*. Vol. 1. Hong Kong, pp. 2–5. URL: http://mlsp.cs.cmu.edu/people/rsingh/papers_old/icassp03-sphinx4_2.pdf (cit. on pp. 34, 181).

Law, Wai Ling, Olga Dmitrieva, and Alexander Francis (2020). "Convergence of L1 and L2 Speech Rhythm in Cantonese-English Bilingual Speakers". In: *Speech Prosody*. Tokyo, Japan, pp. 547–550. DOI: 10.21437/SpeechProsody.2020-112 (cit. on p. 22).

Le Maguer, Sébastien and Ingmar Steiner (2017). "The "Uprooted" MaryTTS Entry for the Blizzard Challenge 2017". In: *Blizzard Challenge*. Stockholm, Sweden. URL: http://festvox.org/blizzard/bc2017/MaryTTS_Blizzard2017.pdf (cit. on p. 182).

Lebeuf, Carlene, Margaret-Anne Storey, and Alexey Zagalsky (2017). "How Software Developers Mitigate Collaboration Friction with Chatbots". In: *arXiv preprint*. URL: https://arxiv.org/pdf/1702.07011.pdf (cit. on p. 39).

Lehnert-LeHouillier, Heike, Susana Terrazas, Steven Sandoval, and Rachel Boren (2020). "The Relationship Between Prosodic Ability and Conversational Prosodic Entrainment". In: *Speech Prosody*, pp. 769–773. DOI: 10.21437/SpeechProsody.2020-157 (cit. on p. 22).

Leong, Victoria, Elizabeth Byrne, Kaili Clackson, Stanimira Georgieva, Sarah Lam, and Sam Wass (2017). "Speaker Gaze Increases Information Coupling Between Infant and Adult Brains". In: *Proceedings of the National Academy of Sciences* 114.50, pp. 13290–13295. DOI: 10.17863/CAM.24090 (cit. on p. 4).

Levin, Esther, Roberto Pieraccini, and Wieland Eckert (2000). "A Stochastic Model of Human-Machine Interaction for Learning Dialog Strategies". In: *IEEE Transactions on Speech and Audio Processing* 8.1, pp. 11–23. DOI: 10.1109/89.817450 (cit. on p. 28).

Levitan, Rivka (2013). "Entrainment in Spoken Dialogue Systems: Adopting, Predicting and Influencing User Behavior". In: *NAACL HLT Student Research Workshop*,

pp. 84–90. URL: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.378.6069&rep=rep1&type=pdf (cit. on pp. 4, 16, 60).

— (2014). "Acoustic-Prosodic Entrainment in Human-Human and Human-Computer Dialogue". PhD thesis. New York, NY, USA: Columbia University. DOI: 10.7916/D8GT5KCH (cit. on p. 29).

— (2020). "Developing an Integrated Model of Speech Entrainment". In: DOI: 10.24963/ijcai.2020/727 (cit. on p. 48).

Levitan, Rivka, Stefan Benus, Ramiro H Gálvez, Agustín Gravano, Florencia Savoretti, Marian Trnka, Andreas Weise, and Julia Hirschberg (Sept. 2016). "Implementing Acoustic-Prosodic Entrainment in a Conversational Avatar". In: *Interspeech*. San Francisco, CA, USA, pp. 1166–1170. DOI: 10.21437/Interspeech.2016-985 (cit. on pp. 6, 46, 49, 104, 191).

Levitan, Rivka, Agustín Gravano, Laura Willson, Štefan Beňuš, Julia Hirschberg, and Ani Nenkova (2012). "Acoustic-Prosodic Entrainment and Social Behavior". In: *Human Language Technologies*. North American Chapter of the Association for Computational Linguistics, pp. 11–19. URL: https://www.aclweb.org/anthology/N12-1002.pdf (cit. on p. 22).

Levitan, Rivka and Julia Hirschberg (2011). "Measuring Acoustic-Prosodic Entrainment with Respect to Multiple Levels and Dimensions". In: *Interspeech*. Florence, Italy, pp. 3081–3084. URL: http://www.isca-speech.org/archive/interspeech_2011/i11_3081.html (cit. on pp. 4, 14, 16, 19, 22, 55, 110).

Lewandowski, Natalie (2012). "Talent in Non-Native Phonetic Convergence". PhD thesis. University of Stuttgart. URL: http://elib.uni-stuttgart.de/bitstream/11682/2875/1/Lewandowski.pdf (cit. on p. 21).

Lewandowski, Natalie and Matthias Jilka (2019). "Phonetic Convergence, Language Talent, Personality and Attention". In: *Frontiers in Communication* 18. DOI: 10.3389/fcomm.2019.00018 (cit. on p. 22).

Li, Jiwei, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky (2016). "Deep Reinforcement Learning for Dialogue Generation". In: *arXiv preprint*. URL: https://arxiv.org/pdf/1606.01541.pdf (cit. on p. 36).

Lin, Jessica, Eamonn Keogh, Li Wei, and Stefano Lonardi (2007). "Experiencing SAX: A Novel Symbolic Representation of Time Series". In: *Data Mining and Knowledge Discovery* 15.2, pp. 107–144. DOI: 10.1007/s10618-007-0064-z (cit. on p. 148).

Lison, Pierre and Casey Kennington (2015). "Developing Spoken Dialogue Systems with the OpenDial Toolkit". In: *Semantics and Pragmatics of Dialogue (SemDial)*. Gothenburg, Sweden, pp. 194–195. URL: http://semdial.org/anthology/Z15-Lison_semdial_0039.pdf (cit. on p. 176).

— (2016). "OpenDial: A toolkit for Developing Spoken Dialogue Systems with Probabilistic Rules". In: *Annual Meeting of the Association for Computational Linguistics*. Berlin, Germany, pp. 67–72. URL: http://aclweb.org/anthology/P16-4012 (cit. on pp. 176, 181).

Local, John (2007). "Phonetic Detail and the Organisation of Talk-in-Interaction". In: *International Congress of Phonetic Sciences (ICPhS)*. URL: https://pdfs.semanticscholar.org/8f73/42a2cec6ee2a8d9bcc9cd004290d87bb5d55.pdf (cit. on pp. 4, 22).

Lopes, José, Maxine Eskenazi, and Isabel Trancoso (2011). "Towards Chosing Better Primes for Spoken Dialog Systems". In: *Automatic Speech Recognition and Understanding*. Waikoloa, HI, USA, pp. 306–311. DOI: 10.1109/ASRU.2011.6163949 (cit. on p. 28).

— (2013). "Automated Two-Way Entrainment to Improve Spoken Dialog System Performance". In: *Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 8372–8376. DOI: 10.1109/ICASSP.2013.6639298 (cit. on p. 16).

Lopes, José DavidÁguas (Dec. 2013). "Lexical Entrainment in Spoken Dialog Systems". PhD thesis. Lisbon, Portugal: Instituto Superior Técnico (cit. on p. 19).

Lubold, Nichola, Heather Pon-Barry, and Erin Walker (2015). "Naturalness and Rapport in a Pitch Adaptive Learning Companion". In: *Automatic Speech Recognition and Understanding (ASRU)*. IEEE, pp. 103–110. DOI: 10.1109/ASRU.2015.7404781 (cit. on p. 48).

MacDorman, Karl F (2006). "Subjective Ratings of Robot Video Clips for Human Likeness, Familiarity, and Eeriness: An Exploration of the Uncanny Valley". In: *Toward Social Mechanisms of Android Science*. Vancouver, Canada: ICCS/CogSci, pp. 26–29. URL: http://www.macdorman.com/kfm/writings/pubs/MacDorman2006SubjectiveRatings.pdf (cit. on p. 192).

Marino, José B, Rafael E Banchs, Josep M Crego, Adrià de Gispert, Patrik Lambert, José AR Fonollosa, and Marta R Costa-jussà (2006). "N-Gram-Based Machine Translation". In: *Computational Linguistics* 32.4, pp. 527–549. DOI: 10.1162/coli.2006.32.4.527 (cit. on p. 147).

Martin, W. Steve (2017). "6 Reasons Salespeople Win or Lose a Sale". In: *Harvard Business Review*. URL: https://hbr.org/2017/06/6-reasons-salespeople-win-or-lose-a-sale (cit. on p. 54).

Marwan, Norbert, M Carmen Romano, Marco Thiel, and Jürgen Kurths (2007). "Recurrence Plots for the Analysis of Complex Systems". In: *Physics Reports* 438.5-6, pp. 237–329. DOI: 10.1016/j.physrep.2006.11.001 (cit. on pp. 63, 66).

Marwan, Norbert, Marco Thiel, and Norbert R Nowaczyk (2002). "Cross Recurrence Plot Based Synchronization of Time Series". In: *arXiv preprint*. DOI: 10.5194/npg-9-325-2002 (cit. on p. 61).

Mehr, Hila (Aug. 2017). "Artificial Intelligence for Citizen Services and Government". In: *Ash Center for Democratic Governance and Innovation*, pp. 1–12. URL: https://ash.harvard.edu/files/ash/files/artificial_intelligence_for_citizen_services.pdf (cit. on p. 56).

Messum, Piers (2007). "Mirroring, Not Imitation, for the Early Learning of L1 Pronunciation". In: *Journal of the Acoustical Society of America* 122.5, p. 2997. DOI: 10.1121/1.2942701 (cit. on p. 18).

Messum, Piers and Ian S Howard (2015). "Creating the Cognitive Form of Phonological Units: The Speech Sound Correspondence Problem in Infancy Could be Solved by Mirrored Vocal Interactions Rather Than by Imitation". In: *Journal of Phonetics* 53, pp. 125–140. DOI: 10.1016/j.wocn.2015.08.005 (cit. on p. 18).

Meyer, David E and Roger W Schvaneveldt (1971). "Facilitation in Recognizing Pairs of Words: Evidence of a Dependence Between Retrieval Operations". In: *Journal of Experimental Psychology* 90.2, p. 227. DOI: 10.1037/h0031564 (cit. on p. 17).

Meyer, Leonard B. (2008). *Emotion and Meaning in Music*. University of Chicago Press (cit. on p. 78).

Michalsky, Jan and Heike Schoormann (2017). "Pitch Convergence as an Effect of Perceived Attractiveness and Likability". In: *Interspeech*. Stockholm, Sweden, pp. 2253–2256. DOI: 10.21437/Interspeech.2017-1520 (cit. on p. 22).

Mitterer, Holger and Jochen Müsseler (2013). "Regional Accent Variation in the Shadowing Task: Evidence for a Loose Perception-Action Coupling in Speech". In: *Attention, Perception, & Psychophysics* 75.3, pp. 557–575. DOI: 10.3758/s13414-012-0407-8 (cit. on p. 88).

Molino, Jean (Oct. 2000). *Toward an Evolutionary Theory of Music and Language*. The MIT Press (cit. on p. 77).

Mori, Masahiro (1970). "The Uncanny Valley". In: *Energy* 7. in Japanese, pp. 33–35. DOI: 10.1109/MRA.2012.2192811 (cit. on p. 192).

Nardo, Davide and Susanne Reiterer (2009). "Musicality and Phonetic Language Aptitude". In: *Language Talent and Brain Activity*. Ed. by Grzegorz Dogil and Susanne Reiterer. Berlin: Mouton de Gruyter Berlin, pp. 213–256. DOI: 10.1515/9783110215496 (cit. on p. 99).

Nass, Clifford and Youngme Moon (2000). "Machines and Mindlessness: Social Responses to Computers". In: *Journal of Social Issues* 56.1, pp. 81–103. DOI: 10.1111/0022-4537.00153 (cit. on pp. 27, 88, 123).

Nass, Clifford, Jonathan Steuer, and Ellen R Tauber (1994). "Computers are Social Actors". In: *conference on Human Factors in Computing Systems (SIGCHI)*, pp. 72–78. DOI: 10.1145/191666.191703 (cit. on pp. 27, 88, 123).

Natale, Michael (1975). "Convergence of Mean Vocal Intensity in Dyadic Communication as a Function of Social Desirability". In: *Journal of Personality and Social Psychology* 32.5, pp. 790–804. DOI: 10.1037/0022-3514.32.5.790 (cit. on pp. 22, 110).

Nenkova, Ani, Agustin Gravano, and Julia Hirschberg (2008). "High Frequency Word Entrainment in Spoken Dialogue". In: *Human Language Technologies*. Columbus, OH, USA, pp. 169–172. URL: http://aclweb.org/anthology/P08-2043 (cit. on p. 22).

Nielsen, Kuniko (2011). "Specificity and Abstractness of VOT Imitation". In: *Journal of Phonetics* 39.2, pp. 132–142. DOI: 10.1016/j.wocn.2010.12.007 (cit. on p. 22).

Niesler, Thomas R and Philip C Woodland (1996). "A Variable-Length Category-Based N-Gram Language Model". In: *International Conference on Acoustics, Speech, and Signal Processing*. Vol. 1. IEEE, pp. 164–167. DOI: 10.1109/ICASSP.1996.540316 (cit. on p. 147).

Noy, Pinhas (1999). *The Psychoanalysis of Art and Creativity*. Modan (cit. on p. 99).

Ohala, John J (1989). "Sound Change Is Drawn from a Pool of Synchronic Variation". In: *Language Change: Contributions to the Study of its Causes*, pp. 173–198. DOI: 10.1515/9783110853063.173 (cit. on p. 23).

— (1990). "The Phonetics And Phonology of Aspects of Assimilation". In: *Laboratory Phonology* 1, pp. 258–275. DOI: 10.1017/CBO9780511627736.014 (cit. on pp. 17, 23).

— (1993). “The Phonetics Of Sound Change”. In: *Historical Linguistics: Problems and Perspectives*, pp. 237–278 (cit. on p. 23).

Okada, Brooke M, Lorin Lachs, and Benjamin Boone (2012). “Interpreting Tone of Voice: Musical Pitch Relationships Convey Agreement in Dyadic Conversation”. In: *The Journal of the Acoustical Society of America* 132.3, EL208–EL214. DOI: 10.1121/1.4742316 (cit. on p. 99).

Orlob, Chris (2017a). “The Science of Winning Sales Conversations”. In: URL: https://www.gong.io/blog/winning-sales-conversations/ (cit. on pp. 54, 56).

— (2017b). “This Is What Separates Your Star Reps from the Rest of The Team”. In: URL: https://www.gong.io/blog/this-is-what-separates-yourstar-reps-from-the-rest-of-the-team (cit. on p. 56).

— (2018). *9 Secret Elements of Highly Effective Sales Conversations*. URL: https://www.gong.io/blog/elements-of-effective-sales-conversations/ (cit. on pp. 54, 85).

— (2019). “ROI In Sales Is Dead. Great Salespeople Are Doing This Now Instead”. In: URL: https://www.gong.io/blog/sales-roi/ (cit. on pp. 60, 73).

Osborne, Joe (July 2016). *Why 100 Million Monthly Cortana Users on Windows 10 Is a Big Deal*. URL: https://www.techradar.com/news/software/operating-systems/why-100-million-monthly-cortana-users-could-be-a-bigger-deal-than-350-million-windows-10-installs-1325146 (cit. on p. 37).

Oviatt, Sharon, Courtney Darves, and Rachel Coulston (2004). “Toward Adaptive Conversational Interfaces: Modeling Speech Convergence with Animated Personas”. In: *ACM Transactions on Computer-Human Interaction* 11.3, pp. 300–328. DOI: 10.1145/1017494.1017498 (cit. on p. 41).

Pace-Sigge, Michael (2013). “The Concept of Lexical Priming in The Context of Language Use”. In: *ICAME Journal* 37, pp. 149–173 (cit. on p. 17).

Papoušek, Mechthild, Hanuš Papoušek, and David Symmes (1991). “The Meanings of Melodies in Motherese in Tone and Stress Languages”. In: *Infant Behavior and Development* 14.4, pp. 415–440. DOI: https://doi.org/10.1016/0163-6383(91)90031-M (cit. on p. 79).

Pardo, Jennifer (2013). “Measuring Phonetic Convergence in Speech Production”. In: *Frontiers in Psychology* 4.559, pp. 1–5. DOI: 10.3389/fpsyg.2013.00559 (cit. on p. 21).

Pardo, Jennifer S, Rachel Gibbons, Alexandra Suppes, and Robert M Krauss (2012). "Phonetic Convergence in College Roommates". In: *Journal of Phonetics* 40.1, pp. 190–197. DOI: 10.1016/j.wocn.2011.10.001 (cit. on p. 78).

Pardo, Jennifer S, Isabel Cajori Jay, and Robert M Krauss (2010). "Conversational Role Influences Speech Imitation". In: *Attention, Perception, & Psychophysics* 72.8, pp. 2254–2264. DOI: 10.3758/bf03196699 (cit. on p. 21).

Pardo, Jennifer S, Adelya Urmanche, Sherilyn Wilman, and Jaclyn Wiener (2017). "Phonetic Convergence Across Multiple Measures and Model Talkers". In: *Attention, Perception, & Psychophysics* 79.2, pp. 637–659. DOI: 10.3758/s13414-016-1226-0 (cit. on p. 38).

Pardo, Jennifer S, Adelya Urmanche, Sherilyn Wilman, Jaclyn Wiener, Nicholas Mason, Keagan Francis, and Melanie Ward (2018). "A Comparison of Phonetic Convergence in Conversational Interaction and Speech Shadowing". In: *Journal of Phonetics* 69, pp. 1–11. DOI: 10.1016/j.wocn.2018.04.001 (cit. on p. 76).

Pardo, Jennifer S. (Apr. 2006). "On Phonetic Convergence During Conversational Interaction". In: *Journal of the Acoustical Society of America* 119.4, pp. 2382–2393. DOI: 10.1121/1.2178720 (cit. on pp. 14, 21).

Parent, Gabriel and Maxine Eskenazi (2010). "Lexical Entrainment of Real Users in the Let's Go Spoken Dialog System". In: *Interspeech*. Makuhari, Chiba, Japan, pp. 3018–3021. URL: http://www.isca-speech.org/archive/interspeech_2010/i10_3018.html (cit. on pp. 4, 28).

Parrill, Fey and Irene Kimbara (2006). "Seeing and Hearing Double: The Influence of Mimicry in Speech and Gesture on Observers". In: *Journal of Nonverbal Behavior* 30.4, p. 157. DOI: 10.1007/s10919-006-0014-2 (cit. on p. 18).

Peterson, Mark (2005). "Learning Interaction in an Avatar-Based Virtual Environment: A Preliminary Study". In: *PacCALL* 1.1, pp. 29–40. DOI: 10.1111/j.1467-8535.2009.00991.x (cit. on p. 40).

Pickering, Martin J and Simon Garrod (2013). "An Integrated Theory of Language Production and Comprehension". In: *Behavioral and Brain Sciences* 36.4, pp. 329–347. DOI: 10.1017/S0140525X12001495 (cit. on p. 13).

— (2004). "Toward a Mechanistic Psychology of Dialogue". In: *Behavioral and Brain Sciences* 27.2, pp. 169–190. DOI: 10.1017/S0140525X04000056 (cit. on pp. 13, 17, 21, 60).

Pilato, Giovanni, Giorgio Vassallo, Agnese Augello, Maria Vasile, and Salvatore Gaglio (2005). "Expert Chat-Bots for Cultural Heritage". In: *Intelligenza Artificiale* 2.2, pp. 25–31. URL: https://s3.amazonaws.com/academia.edu.documents/42545699/02_pilato.pdf?AWSAccessKeyId=AKIAIWOWYYGZ2Y53UL3A&Expires=1508414198&Signature=pm%2B7jTsc2DrP8cmPslgWImG%2FgNA%3D&response-content-disposition=inline%3B%20filename%3DExpert_chat-bots_for_cultural_heritage.pdf (cit. on p. 39).

Platt, John (1998). *Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines*. Tech. rep. MSR-TR-98-14. Microsoft, pp. 1–21 (cit. on p. 186).

Platt, John C (Feb. 1999). "Fast Training of Support Vector Machines Using Sequential Minimal Optimization". In: *Advances in Kernel Methods*. Ed. by Christopher J. C. Burges, Bernhard Schölkopf, and Alexander J. Smola. MIT Press, pp. 185–208. DOI: 10.1109/ISKE.2008.4731075 (cit. on p. 186).

Porzel, Robert, Annika Scheffler, and Rainer Malaka (2006). "How Entrainment Increases Dialogical Efficiency". In: *Effective Multimodal Dialogue Interfaces*. Sydney, Australia (cit. on p. 41).

Povey, Daniel, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, and Petr Schwarz (2011). "The Kaldi Speech Recognition Toolkit". In: *Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society. DOI: 10.1.1.468.3637 (cit. on p. 34).

Prenger, Ryan, Raffael Valle, and Bryan Catanzaro (May 2019). "WaveGlow: A Flow-based Generative Network for Speech Synthesis". In: *ICASSP*. Brighton, UK, pp. 3617–3621. DOI: 10.1109/ICASSP.2019.8683143 (cit. on p. 171).

Press, W.H., S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery (1992). "Numerical Recipes in C – The Art of Scientific Computing". In: *The Mathematical Gazette* 73.464 (cit. on p. 171).

Putman, William B and Richard L Street (1984). "The Conception and Perception of Noncontent Speech Performance: Implications for Speech-accommodation Theory". In: *International Journal of the Sociology of Language* 1984.46, pp. 97–114. DOI: 10.1515/ijsl.1984.46.97 (cit. on p. 22).

Rácz, Péter, Clay Beckner, Jennifer B. Hay, and Janet B. Pierrehumbert (2020). "Morphological Convergence as On-Line Lexical Analogy". In: *Language*. URL: http://www.phon.ox.ac.uk/jpierrehumbert/publications/Racz_etal_Language_SI.pdf (cit. on p. 19).

Rahimi, Zahra and Diane Litman (2018). "Weighting Model Based on Group Dynamics to Measure Convergence in Multi-party Dialogue". In: *SIGdial*. Association for Computational Linguistics, pp. 385–390. URL: http://www.aclweb.org/anthology/W18-5046 (cit. on pp. 25, 60).

Rahimi, Zahra, Diane Litman, and Susannah Paletz (2019). "Acoustic-Prosodic Entrainment in Multi-party Spoken Dialogues: Does Simple Averaging Extend Existing Pair Measures Properly?" In: *Advanced Social Interaction with Agents*. Springer, pp. 169–177. DOI: 10.21437/Interspeech.2017-1568. URL: http://people.cs.pitt.edu/~litman/RahimiChapter.pdf (cit. on p. 105).

Raux, Antoine, Brian Langner, Dan Bohus, Alan W Black, and Maxine Eskenazi (2005). "Let's Go Public! Taking a Spoken Dialog System to the Real World". In: *Interspeech*. Lisbon, Portugal, pp. 885–888. URL: http://www.isca-speech.org/archive/interspeech_2005/i05_0885.html (cit. on p. 28).

Raveh, Eran, Iona Gessinger, Sébastien Le Maguer, Ingmar Steiner, and Bernd Möbius (Mar. 2017a). "Investigating Phonetic Convergence in a Shadowing Experiment with Synthetic Stimuli". In: *Electronic Speech Signal Processing (ESSV)*. Ed. by Jürgen Trouvain, Ingmar Steiner, and Bernd Möbius. Saarbrücken, Germany, pp. 254–261. URL: http://essv2017.coli.uni-saarland.de/pdfs/Raveh.pdf (cit. on pp. 9, 93).

Raveh, Eran, Ingo Siegert, Ingmar Steiner, Iona Gessinger, and Bernd Möbius (Sept. 2019a). "Three's a Crowd? Effects of a Second Human on Vocal Accommodation with a Voice Assistant". In: *Interspeech*. Graz, Austria, pp. 4005–4009. DOI: 10.21437/Interspeech.2019-1825 (cit. on pp. 9, 98, 105).

Raveh, Eran and Ingmar Steiner (Sept. 2017a). "Automatic Analysis of Segmental Features in a Real-Time Phonetic Convergence Pipeline". In: *Phonetik und Phonologie im deutschsprachigen Raum*. Berlin, Germany. URL: http://www.coli.uni-saarland.de/~raveh/assets/pdfs/Raveh2017PundP.pdf (cit. on p. 10).

— (Aug. 2017b). "Phonetic Adaptation Module for Spoken Dialogue Systems". In: *Semantics and Pragmatics of Dialogue (SemDial)*. Saarbrücken, Germany, pp. 170–171.

URL: http://semdial.org/anthology/Z17-Raveh_semdial_0027.pdf (cit. on pp. 10, 162, 176).

Raveh, Eran, Ingmar Steiner, Iona Gessinger, and Bernd Möbius (Sept. 19, 2018). "Studying Mutual Phonetic Influence With a Web-Based Spoken Dialogue System". In: *Speech and Computer (Specom)*. Ed. by Alexey Karpov, Oliver Jokisch, and Rodmonga Potapova. Vol. 11096. Lecture Notes in Artificial Intelligence. Springer, pp. 552–562. DOI: 10.1007/978-3-319-99579-3_57. URL: https://arxiv.org/abs/1809.04945 (cit. on pp. 10, 59, 110).

— (Sept. 2019b). "Analyzing Phonetic Accommodation in Human-Human and Human-Computer Interactions". In: *Phonetik und Phonologie im deutschsprachigen Raum*. Düsseldorf, Germany (cit. on p. 10).

Raveh, Eran, Ingmar Steiner, and Bernd Möbius (Aug. 2017b). "A Computational Model for Phonetically Responsive Spoken Dialogue Systems". In: *Interspeech*. Stockholm, Sweden, pp. 884–888. DOI: 10.21437/Interspeech.2017-1042 (cit. on pp. 10, 46, 135).

Raveh, Eran, Ingmar Steiner, Ingo Siegert, Iona Gessinger, and Bernd Möbius (Mar. 2019c). "Comparing Phonetic Changes in Computer-Directed and Human-Directed Speech". In: *Electronic Speech Signal Processing (ESSV)*. Dresden, Germany: TUDpress, pp. 42–49 (cit. on pp. 9, 105).

Raveh, Eran, Maya Twig, Bernd Möbius, and Oded Zehavi (May 2020). "Prosodic Alignments in Shadowed Singing of Familiar and Novel Music". In: *Speech Prosody*. Tokyo, Japan, pp. 606–610. DOI: 10.21437/SpeechProsody.2020-124. URL: http://dx.doi.org/10.21437/SpeechProsody.2020-124 (cit. on pp. 9, 81).

Reitter, David, Frank Keller, and Johanna D Moore (2006). "Computational Modelling of Structural Priming in Dialogue". In: *Human Language Technology Conference*. Association for Computational Linguistics, pp. 121–124. DOI: 10.3115/1614049.1614080 (cit. on p. 17).

Rothkrantz, Leon JM, Pascal Wiggers, Jan-Willem A Van Wees, and Robert J van Vark (2004). "Voice Stress Analysis". In: *Text, Speech and Dialogue*. Springer, pp. 449–456. DOI: 10.1007/978-3-540-30120-2_57 (cit. on p. 169).

Schröder, Marc, Marcela Charfuelan, Sathish Pammi, and Ingmar Steiner (2011). "Open Source Voice Creation Toolkit for the MARY TTS Platform". In: *International Speech Communication Association*. DOI: 10.1.1.228.1771 (cit. on p. 36).

Schröder, Marc and Jürgen Trouvain (2003). "The German Text-to-Speech Synthesis System MARY: A Tool for Research, Development and Teaching". In: *International Journal of Speech Technology* 6.4, pp. 365–377. DOI: 10.1023/A:1025708916924 (cit. on p. 182).

Schvaneveldt, RW and David E Meyer (1973). *Retrieval and Comparison Processes in Semantic Memory*. Vol. 4. New York: Academic Press (cit. on p. 17).

Schweitzer, Antje and Natalie Lewandowski (2013). "Convergence of Articulation Rate in Spontaneous Speech". In: *Interspeech*. Lyon, France, pp. 525–529. DOI: 10.1.1.394.4806 (cit. on pp. 78, 110, 123).

Schweitzer, Antje, Natalie Lewandowski, and Daniel Duran (2017a). "Social Attractiveness in Dialogs". In: *Interspeech*, pp. 2243–2247. DOI: 10.21437/Interspeech.2017-833 (cit. on pp. 38, 60).

Schweitzer, Antje and Michael Walsh (2016). "Exemplar Dynamics in Phonetic Convergence of Speech Rate". In: *Interspeech*. San Francisco, CA, USA, pp. 2100–2104. DOI: 10.21437/Interspeech.2016-373 (cit. on pp. 22, 45).

Schweitzer, Katrin, Michael Walsh, and Antje Schweitzer (2017b). "To See or Not to See: Interlocutor Visibility and Likeability Influence Convergence in Intonation". In: *Interspeech*. Stockholm, Sweden, pp. 919–923. DOI: 10.21437/Interspeech.2017-1248 (cit. on p. 22).

Shapiro, Stuart C (1992). *Encyclopedia of Artificial Intelligence*. John Wiley. DOI: 10.1017/S0263574700016489 (cit. on p. 42).

Shawar, Bayan Abu and Eric Atwell (2007). "Chatbots: Are They Really Useful?" In: *LDV Forum*. Vol. 22. 1, pp. 29–49. URL: https://www.researchgate.net/profile/Eric_Atwell/publication/220046725_Chatbots_Are_they_Really_Useful/links/00b7d518ab7329add2000000/Chatbots-Are-they-Really-Useful.pdf (cit. on p. 39).

Shen, Jonathan, Ruoming Pang, Ron J Weiss, Mike Schuster, Navdeep Jaitly, Zongheng Yang, Zhifeng Chen, Yu Zhang, Yuxuan Wang, and Rj Skerrv-Ryan (Apr. 2018). "Natural TTS Synthesis by Conditioning Wavenet on Mel Spectrogram Predictions". In: *Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, pp. 4779–4783. DOI: 10.1109/ICASSP.2018.8461368 (cit. on pp. 35, 171).

Shepard, Carolyn A., Howard Giles, and Beth A. Le Poire (2001). "Communication Accommodation Theory". In: *The New Handbook of Language and Social Psychology*. Ed. by W. Peter Robinson and Howard Giles. Wiley, pp. 33–56 (cit. on p. 40).

Shockley, Kevin, Laura Sabadini, and Carol A Fowler (2004). "Imitation in Shadowing Words". In: *Perception & Psychophysics* 66.3, pp. 422–429. DOI: 10.3758/bf03194890 (cit. on pp. 21, 76).

Shriberg, Elizabeth, Andreas Stolcke, and Suman Ravuri (Aug. 2013). "Addressee Detection for Dialog Systems Using Temporal and Spectral Dimensions of Speaking Style". In: *Interspeech*. Lyon, France, pp. 2559–2563. URL: https://www.isca-speech.org/archive/interspeech_2013/i13_2559.html (cit. on p. 106).

Siegert, Ingo, Julia Krüger, Olga Egorow, Jannik Nietzold, Ralph Heinemann, and Alicia Lotz (May 2018). "Voice Assistant Conversation Corpus (VACC): A Multi-Scenario Dataset for Addressee Detection in Human-Computer-Interaction using Amazon's Alexa". In: *Workshop on Language and Body in Real Life & Multimodal Corpora*. Miyazaki, Japan. URL: http://lrec-conf.org/workshops/lrec2018/W20/pdf/13_W20.pdf (cit. on pp. 106, 107).

Silber-Varod, Vered (2018). "Is Human-Human Spoken Interaction Manageable? The Emergence of the Concept "Conversation Intelligence"". In: *Online Journal of Applied Knowledge Management (OJAKM)*. DOI: 10.36965/OJAKM.2018.6(1)1-14 (cit. on pp. 56, 57).

Silber-Varod, Vered, Anat Lerner, and Oliver Jokisch (Sept. 2018). "Prosodic Plot of Dialogues: A Conceptual Framework to Trace Speakers' Role". In: *International Conference on Speech and Computer*, pp. 636–645. DOI: 10.1007/978-3-319-99579-3_65 (cit. on pp. 46, 109).

Simonet, Miquel (2011). "Intonational Convergence in Language Contact: Utterance-Final f0 Contours in Catalan-Spanish Early Bilinguals". In: *Journal of the International Phonetic Association* 41.2, pp. 157–184. DOI: 10.1017/S0025100311000120 (cit. on p. 78).

Singh, Avinash Kumar and Gora Chand Nandi (2016). "NAO Humanoid Robot: Analysis of Calibration Techniques for Robot Sketch Drawing". In: *Robotics and Autonomous Systems* 79, pp. 108–121. DOI: 10.1016/j.robot.2016.01.009 (cit. on p. 39).

Skantze, Gabriel and David Schlangen (2009). "Incremental Dialogue Processing in a Micro-Domain". In: *European Chapter of the Association for Computational Linguistics (EACL)*. DOI: 10.3115/1609067.1609150 (cit. on p. 42).

Smith, Caroline L (2007). “Prosodic Accommodation by French Speakers to a Non-Native Interlocutor”. In: *International Congress of Phonetic Sciences*, pp. 313–348 (cit. on p. 21).

Staum Casasanto, Laura, Kyle Jasmin, and Daniel Casasanto (2010). “Virtually Accommodating: Speech Rate Accommodation to a Virtual Interlocutor”. In: *Cognitive Science Society (CogSci)*. Cognitive Science Society, pp. 127–132. URL: https://pure.mpg.de/rest/items/item_458220/component/file_529234/content (cit. on pp. 40, 122).

Steiner, Ingmar, Marc Schröder, and Annette Klepp (2013). “The PAVOQUE Corpus as a Resource for Analysis and Synthesis of Expressive Speech”. In: *Phonetik und Phonologie im deutschsprachigen Raum* 9 (cit. on p. 171).

Stolcke, Andreas (2002). “SRILM – An Extensible Language Modeling Toolkit”. In: *Interspeech*. Denver, CO, USA, pp. 901–904. URL: http://www.isca-speech.org/archive/icslp_2002/i02_0901.html (cit. on p. 182).

Street, Alison, Susan Young, Johannella Tafuri, and Beatriz Ilari (2003). “Mothers’ Attitudes to Singing to Their Infants”. In: *ESCOM*, pp. 628–631. URL: https://www.epos.uni-osnabrueck.de/books/k/klww003/pdfs/099_Street_Proc.pdf (cit. on p. 79).

Suzuki, Noriko, Yugo Takeuchi, Kazuo Ishii, and Michio Okada (2003). “Effects of Echoic Mimicry Using Hummed Sounds on Human-Computer Interaction”. In: *Speech Communication* 40.4, pp. 559–573. DOI: 10.1016/S0167-6393(02)00180-2 (cit. on p. 48).

Sweet, Henry (1874). “The History or English Sounds”. In: *Transactions of the Philological Society* 15.1, pp. 461–623. DOI: 10.1111/j.1467-968X.1874.tb00881.x (cit. on p. 23).

Traum, David and Jeff Rickel (2002). “Embodied Agents for Multi-Party Dialogue in Immersive Virtual Worlds”. In: *Autonomous Agents and Multiagent Systems*, pp. 766–773. DOI: 10.1145/544862.544922 (cit. on p. 105).

Trehub, Sandra E and Anna M Unyk (1991). “Music Prototypes in Developmental Perspective”. In: *Psychomusicology: A Journal of Research in Music Cognition* 10f.2, p. 73. DOI: 10.1037/h0094140 (cit. on p. 79).

Trehub, Sandra E, Anna M Unyk, and Laurel J Trainor (1993). “Maternal Singing in Cross-Cultural Perspective”. In: *Infant Behavior and Development* 16.3, pp. 285–295. DOI: 10.1016/0163-6383(93)80036-8 (cit. on p. 80).

Tsang, Grace Ji Yan, Edmund L Dana, Morwaread M Farbood, and Susannah V Levi (2018). "Musical Training and the Perception of Phonetic Detail in a Shadowing Task". In: *The Journal of the Acoustical Society of America* 143.3, pp. 1922–1922. DOI: 10.1121/1.5036272 (cit. on p. 99).

Turing, Alan M (Oct. 1950). "Computing Machinery and Intelligence". In: *Mind* 59.236, pp. 433–460. DOI: 10.1093/mind/LIX.236.433 (cit. on p. 42).

Turnhout, Koen van, Jacques Terken, Ilse Bakx, and Berry Eggen (Oct. 2005). "Identifying the Intended Addressee in Mixed Human-Human and Human-Computer Interaction from Non-Verbal Features". In: *Multimodal Interfaces (ICMI)*. Trento, Italy, pp. 175–182. DOI: 10.1145/1088463.1088495 (cit. on p. 106).

Twig, Maya (2016). "Music as a Universal Language: Examining the Effect of Culture on Music Perception at Infancy". MA thesis. University of Haifa. URL: https://haifa.userservices.exlibrisgroup.com/view/delivery/972HAI_MAIN/12137798060002791 (cit. on p. 79).

Unyk, Anna M, Sandra E Trehub, Laurel J Trainor, and E Glenn Schellenberg (1992). "Lullabies and Simplicity: A Cross-Cultural Perspective". In: *Psychology of Music* 20.1, pp. 15–28. DOI: 10.1177/0305735692201002 (cit. on p. 80).

Vapnik, Vladimir (1998). "The Support Vector Method of Function Estimation". In: *Nonlinear Modeling*. Ed. by Johan A. K. Suykens and Joos Vandewalle. Springer, pp. 55–85. DOI: 10.1007/978-1-4615-5703-6-3 (cit. on p. 186).

Vinyals, Oriol and Quoc Le (2015). "A Neural Conversational Model". In: *arXiv preprint*. URL: https://arxiv.org/pdf/1506.05869.pdf (cit. on p. 36).

Walker, Abby and Kathryn Campbell-Kibler (2015). "Repeat What After Whom? Exploring Variable Selectivity in a Cross-Dialectal Shadowing Task". In: *Frontiers in Psychology* 6.546, pp. 1–18. DOI: 10.3389/fpsyg.2015.00546 (cit. on pp. 21, 76).

Wallot, Sebastian and Giuseppe Leonardi (2018). "Analyzing Multivariate Dynamics Using Cross-Recurrence Quantification Analysis (CRQA), Diagonal-Cross-Recurrence Profiles (DCRP), and Multidimensional Recurrence Quantification Analysis (MdRQA) – A tutorial in R". In: *Frontiers in Psychology* 9, p. 2232. DOI: doi:10.3389/fpsyg.2018.02232 (cit. on p. 60).

Wansink, Brian, Mitsuru Shimizu, and Guido Camps (2012). "What Would Batman Eat?: Priming Children to Make Healthier Fast Food Choices". In: *Pediatric Obesity*

7.2, pp. 121–123. URL: http://naturaleater.com/science-articles/what-would%20batman-eat.pdf (cit. on p. 17).

Ward, Nigel and Satoshi Nakagawa (2004). "Automatic User-Adaptive Speaking Rate Selection". In: *International Journal of Speech Technology* 7.4, pp. 259–268. DOI: 10.1023/B:IJST.0000037070.31146.f9 (cit. on p. 48).

Webber Jr, Charles L and Joseph P Zbilut (2005). "Recurrence Quantification Analysis of Nonlinear Dynamical Systems". In: *Tutorials in Contemporary Nonlinear Methods for the Behavioral Sciences* 94, pp. 26–94. DOI: 10.1007/978-3-319-07155-8 (cit. on p. 61).

Weise, Andreas (2017). "Towards a Spoken Dialog System Capable of Acoustic-prosodic Entrainment". PhD thesis. City University of New York (cit. on pp. 5, 41).

Weise, Andreas and Rivka Levitan (2018). "Looking for Structure in Lexical and Acoustic-Prosodic Entrainment Behaviors". In: *Human Language Technologies*. Vol. 2. North American Chapter of the Association for Computational Linguistics, pp. 297–302. DOI: 10.18653/v1/N18-2048 (cit. on p. 21).

Weise, Andreas, Sarah Ita Levitan, Julia Hirschberg, and Rivka Levitan (2019). "Individual Differences in Acoustic-Prosodic Entrainment in Spoken Dialogue". In: *Speech Communication*. DOI: 10.1016/j.specom.2019.10.007 (cit. on p. 45).

Weiss, Michael W, Sandra E Trehub, and E Glenn Schellenberg (2012). "Something in the Way She Sings: Enhanced Memory for Vocal Melodies". In: *Psychological Science* 23.10, pp. 1074–1078. DOI: 10.1177/0956797612442552 (cit. on p. 79).

Weizenbaum, Joseph (1966). "ELIZA – A Computer Program for the Study of Natural Language Communication Between Man and Machine". In: *Communications of the ACM* 9.1, pp. 36–45. DOI: 10.1145/365153.365168 (cit. on p. 39).

Wen, Tsung-Hsien, David Vandyke, Nikola Mrksic, Milica Gasic, Lina M Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young (2016). "A Network-Based End-to-End Trainable Task-Oriented Dialogue System". In: *arXiv preprint*. URL: https://arxiv.org/pdf/1604.04562.pdf (cit. on p. 36).

Wilcoxon, Frank (1945). "Individual Comparisons by Ranking Methods". In: *Biometrics Bulletin* 1.6, pp. 80–83. DOI: 10.2307/3001968 (cit. on pp. 68, 111).

Xia, Zhihua, Rivka Levitan, and Julia Hirschberg (2014). "Prosodic Entrainment in Mandarin and English: A Cross-linguistic Comparison". In: *Speech Prosody*. DOI: 10.21437/SPEECHPROSODY.2014-1 (cit. on pp. 14, 16).

Xiao, Bo, Zac E Imel, David C Atkins, Panayiotis G Georgiou, and Shrikanth S Narayanan (2015). "Analyzing Speech Rate Entrainment and Its Relation to Rherapist Empathy in Drug Addiction Counseling". In: *Interspeech*, pp. 2489–2493. URL: https://sail.usc.edu/publications/files/Xiao-IS151344.pdf (cit. on p. 22).

Xu, Ruifeng, Jiyun Zhou, Bin Liu, Yulan He, Quan Zou, Xiaolong Wang, and Kuo-Chen Chou (2015). "Identification of DNA-Binding Proteins by Incorporating Evolutionary Information into Pseudo Amino Acid Composition Via the Top-N-Gram Approach". In: *Journal of Biomolecular Structure and Dynamics* 33.8, pp. 1720–1730. DOI: 10.1080/07391102.2014.968624 (cit. on p. 147).

Yohn, Denise Lee (2016). "The Best Salespeople Do What the Best Brands Do". In: *Harvard Business Review*. URL: https://hbr.org/2016/08/the-best-salespeople-do-what-the-best-brands-do (cit. on p. 54).

Yoshikawa, Yuichiro, Minoru Asada, Koh Hosoda, and Junpei Koga (2003). "A Constructivist Approach to Infants' Vowel Acquisition Through Mother-Infant Interaction". In: *Connection Science* 15.4, pp. 245–258. DOI: 10.1016/j.wocn.2015.08.005 (cit. on p. 18).

Zbilut, Joseph P, Alessandro Giuliani, and Charles L Webber Jr (1998). "Detecting Deterministic Signals in Exceptionally Noisy Environments Using Cross-Recurrence Quantification". In: *Physics Letters A* 246.1-2, pp. 122–128. DOI: 10.1016/S0375-9601(98)00457-5 (cit. on p. 55).

Zbilut, Joseph P and Charles L Webber Jr (1992). "Embeddings and Delays as Derived from Quantification of Recurrence Plots". In: *Physics letters A* 171.3-4, pp. 199–203. DOI: 10.1016/0375-9601(92)90426-M (cit. on p. 61).

Zen, Heiga and Tomoki Toda (2005). "An Overview of Nitech HMM-Based Speech Synthesis System for Blizzard Challenge 2005". In: *Interspeech*. DOI: 10.1093/ietisy/e90-1.1.325. URL: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.210.5914&rep=rep1&type=pdf (cit. on p. 92).

Zen, Heiga, Keiichi Tokuda, and Alan W. Black (Nov. 2009). "Statistical Parametric Speech Synthesis". In: *Speech Communication* 51.11, pp. 1039–1064. DOI: doi.org/10.1016/j.specom.2009.04.004 (cit. on p. 93).

Zhao, Tiancheng and Maxine Eskenazi (2016). "Towards End-to-End Learning for Dialog State Tracking and Management Using Deep Reinforcement Learning". In: *arXiv preprint*. URL: https://arxiv.org/pdf/1606.02560.pdf (cit. on p. 36).

V

Appendices

Appendix A

Shadowing Experiment Stimuli

Recording Fillers

1. Der Schrank wird heute geliefert. (*The cabinet will be delivered today.*)
2. Wo finde ich ein neues Glas. (*Where do I find a new glass?*)
3. Der Markt findet donnerstags statt. (*The market takes place on Thursday.*)
4. Sie wirkt recht gut informiert. (*She seems to be very well informed.*)
5. Ist das der Weg zu dir nach Hause? (*Is this the way to your home?*)

Baseline Fillers

6. Der Eimer ist aus Plastik. (*The bucket is made of plastik.*)
7. Im Kühlschrank liegt ein Pfirsich. (*There is a peach in the fridge.*)
8. Diese Technik wird noch entwickelt. (*This technique will be further developed.*)
9. Das war sehr höflich von dir. (*That was very nice of you.*)
10. Lena geht heute früher ins Bett. (*Lena goes today early to bed.*)

Experiment Fillers

11. Die Katze weckt mich immer auf. (*The cat always wakes me up.*)

12. Der Kaffee war ja schon kalt. (*The coffee was already cold.*)

13. Wer fliegt heute in den Urlaub? (*Who flies today on vacation?*)

14. Warum regt er sich denn so auf? (*Why is he so upset?*)

15. Das wird ein schönes Geschenk. (*This will be a pretty present.*)

16. Ich hätte gern zwei kleine Brüder. (*I would gladly have two brothers.*)

17. Das Heft war gestern noch da. (*Yesterday the notebook was still here.*)

18. Die Glühbirne ist leider kaputt. (*Unfortunately the light bulb is broken.*)

19. Sucht sich Karin eine neue Arbeit? (*Is Karin looking for a new job?*)

20. Wird die Wohnung noch renoviert? (*Will the apartment be renovated?*)

21. Sara hat eine andere Meinung. (*Sara has another opinion.*)

22. Habt ihr das rote Auto erkannt. (*Have you recognized the red car?*)

23. Ich täusche mich so gut wie nie. (*I never delude myself.*)

24. Keiner glaubt diese Geschichte. (*No one believes this story.*)

25. Kommt Fabian auch zu dem Fest. (*Does Fabian come to the festival as well?*)

[ç] **vs.** [k]

26. Kommt Essig in den Salat? (*Does vinegar come into the salad?*)

27. Der König hält eine Rede. (*The king speaks.*)

28. Kommt Ludwig heute Abend mit? (*Does Ludwig join today evening?*)

29. Es ist ganz schön staubig im Keller. (*It is pretty dusty in the basement.*)

30. Ich bin süchtig nach Schokolade. (*I am addicted to chocolate.*)

[e] **vs.** [ɛ]

31. Die Bestätigung ist für Tanja. (*The confirmation is for Tanja.*)
32. War das Gerät sehr teuer? (*Was the device very expensive?*)
33. Ich mag die Qualität deiner Tasche. (*I like the quality of your bag.*)
34. Der Schädling sieht aber komisch aus. (*The pest looks funny.*)
35. Wie viel Verspätung hat der Zug? (*How much delay does the train have?*)

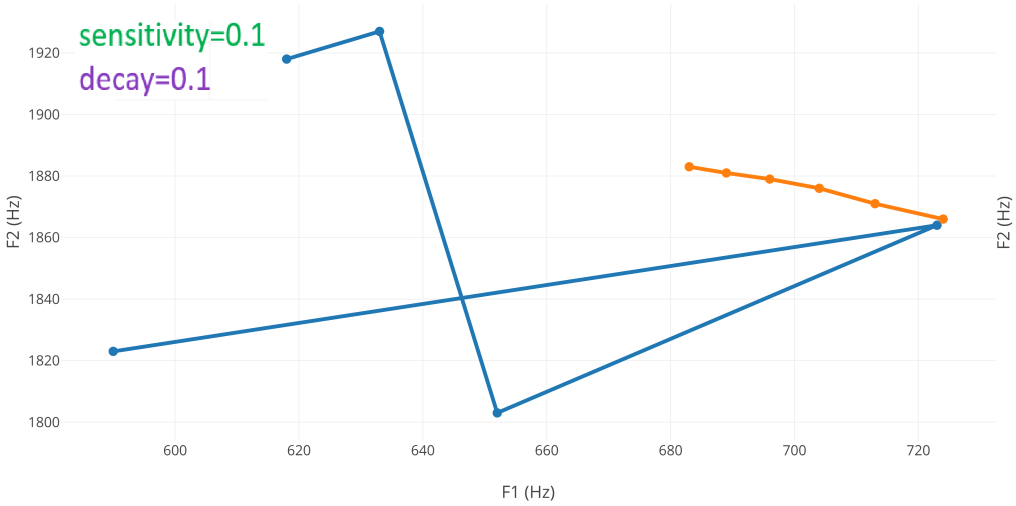
[ən] **vs.** [n̩]

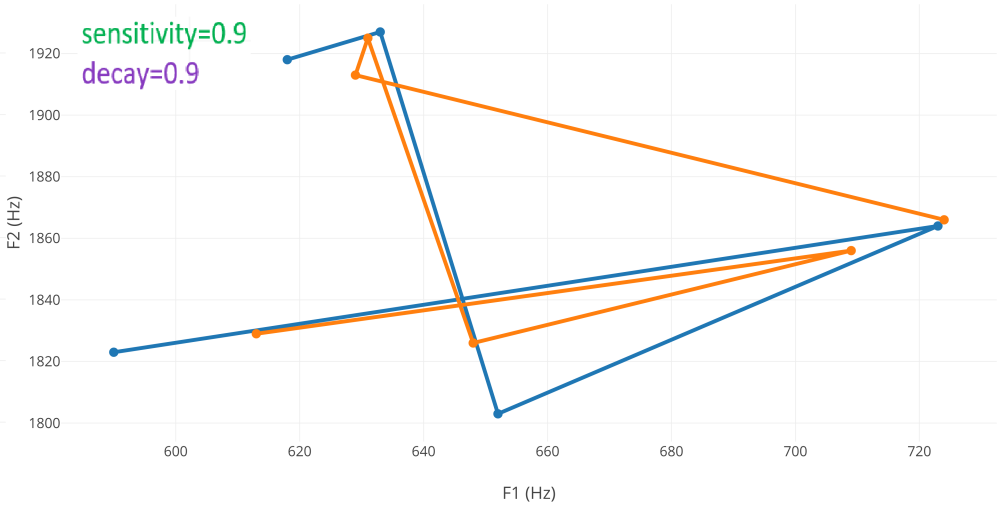
36. Sie begleiten dich zur Taufe. (*They are accompanying you to the baptism.*)
37. Wir besuchen euch bald wieder. (*We will visit you soon again.*)
38. Sind die Küchen immer so groß? (*Are the kitchens always so big?*)
39. Wir reden ohne Unterbrechung. (*We are talking without interruption.*)
40. Sind die Affen denn zutraulich? (*Are the monkeys trustful?*)

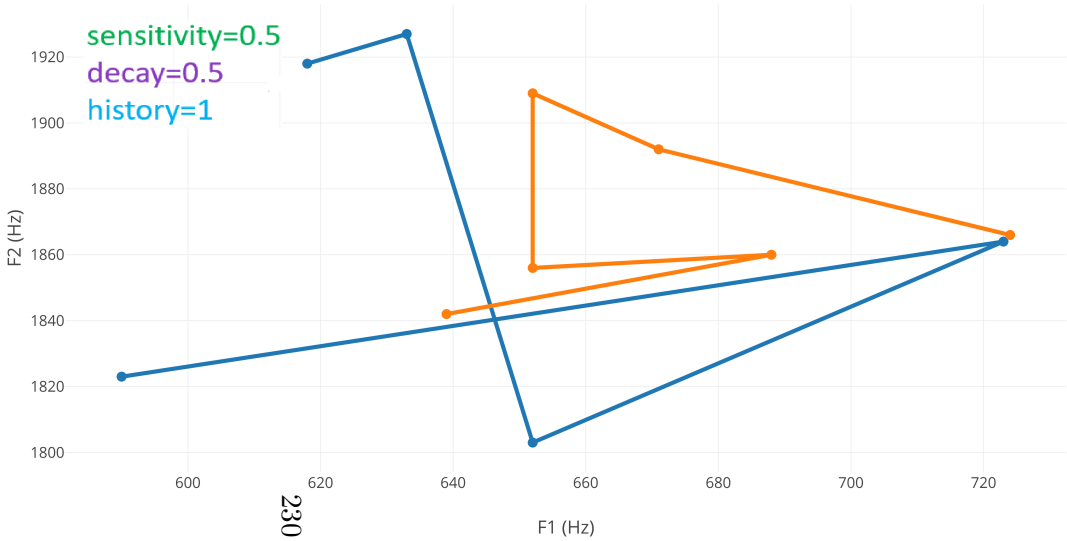
Appendix B

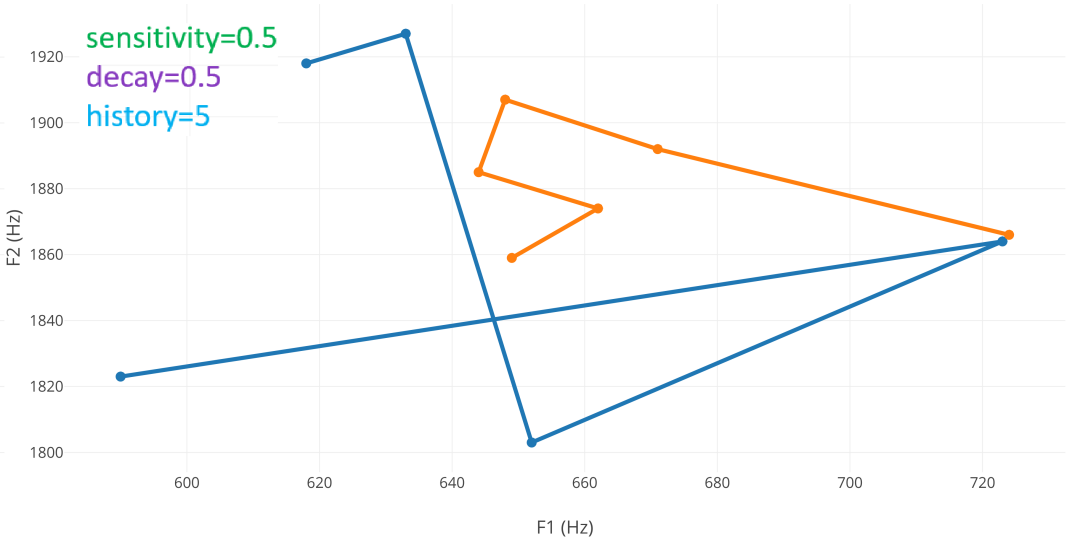
System Visualization Examples

The examples presented here are screenshots of the graphical user interface of the responsive system presented in Chapter 10. These examples compare the state of the system's representation of the [e] vs. [ɛ] feature after processing the same user input but using different parameter values (see Section 7.3 and Table 7.1). It can be seen, for example, how higher sensitivity (top right) leads to faster – but somewhat unstable – convergence process that generally imitates the user's productions. In contrast, the convergence at the top left is too slow to be representative of the user's production. The two bottom examples demonstrate how taking a larger number of previous exemplars into account leads to a more smoothed convergence process toward some global mean (bottom right) as opposed to more rapidly changing productions that follow only the last encountered exemplar (bottom left).


sensitivity=0.1
decay=0.1
F2 (Hz)
F1 (Hz)


sensitivity=0.9
decay=0.9
F2 (Hz)
F1 (Hz)


sensitivity=0.5
decay=0.5
history=1
F2 (Hz)
F1 (Hz)


sensitivity=0.5
decay=0.5
history=5
F2 (Hz)
F1 (Hz)