TECHNISCHE UNIVERSITÄT MÜNCHEN
Forschungs- und Lehreinheit XI
Angewandte Informatik / Kooperative Systeme

Conversational Context for Mobile Notification Management

Florian Alexander Schulze

Vollständiger Abdruck der von der Fakultät für Informatik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften (Dr. rer. nat.)

genehmigten Dissertation.

Vorsitzende:		Univ.-Prof. Gudrun J. Klinker, Ph.D.
Prüfer der Dissertation:	1.	Univ.-Prof. Dr. Johann Schlichter
	2.	Univ.-Prof. Dr. Uwe Baumgarten
	3.	Priv.-Doz. Dr. Georg Groh

Die Dissertation wurde am 14.12.2015 bei der Technischen Universität München eingereicht und durch die Fakultät für Informatik am 08.03.2016 angenommen.

Abstract

This thesis explores if and how identifying the character of face-to-face conversations can help manage proactive actions, especially notifications, on smartphones so that they become less disruptive. We motivate our research with the initial hypothesis that different types of conversation entail a different receptiveness to interruptions. This hypothesis is backed up by surveys and eventually proved in a large-scale user study. We show that the social dimensions *depth/importance* and *formality/goal orientation* of a conversation are strong indicators of receptiveness. We also provide evidence that, contrary to initial belief, *valence* is no indicator by itself. Furthermore, we find that there are types of conversation, small talk for example, in which individuals are even more receptive to notifications than in situations without any verbal social interaction at all. This refutes the assumption currently dominating the literature that the occurrence of a conversation is a strong predictor of unavailability. The study also reveals that 44% of the time a message is received, people are engaged in a conversation, attributing further importance to the consideration of conversational characteristics in notification management. We investigate how and to what degree the character of a conversation can be inferred by technical means – putting strong emphasis on the challenges of mobile scenarios. For that, we study how state-of-the-art methods in speech-based machine learning cope with the effects of noise and signal alterations through the manner the phone is carried by the user. We demonstrate a system that keeps track of conversations in which the user is engaged and that analyzes speech in terms of embedded *affective* and *social* cues. Eventually, we find that information of either kind, derived from audio, improves the accuracy of personal notification preference models for the average individual by more than 20 percent (relative to the baseline set of common context attributes).

Funding for this research project was provided by the German Federal Ministry of Education and Research under the grant ID 01IS12057. The sole responsibility for the report's contents lies with the author.

Kurzfassung

Diese Arbeit folgt der Frage, ob und wie die Erkennung des Charakters von Konversationen helfen kann, Benachrichtigungen auf Smartphones weniger störend zu gestalten. Der Frage liegt die Vermutung zugrunde, dass unterschiedliche Gesprächstypen mit unterschiedlicher Bereitschaft gegenüber Unterbrechungen einhergehen. Diese Hypothese wird durch Umfragen untermauert und letztendlich durch eine großangelegte Studie bewiesen. Wir zeigen, dass die sozialen Dimensionen "Tiefe/Wichtigkeit" und "Formalität/Zielgerichtetheit" starke Indikatoren für Unterbrechbarkeit sind. Entgegen ursprünglicher Annahmen stellt sich Valenz nicht als Indikator heraus. Darüber hinaus zeigt sich, dass die durchschnittliche Unterbrechbarkeit bei bestimmten Gesprächstypen, z.B. Smalltalk, sogar höher ist als in Situationen ohne verbale Interaktion. Dies widerlegt die in der Literatur gegenwärtig vertretene Meinung, dass das Vorhandensein eines Gesprächs eine geringe Unterbrechbarkeit nahelegt. Unsere Studie offenbart, dass Individuen in 44% der Fälle, in denen eine Nachricht empfangen wird, in Gespräche verwickelt sind. Dies unterstreicht wie wichtig die nähere Betrachtung von Gesprächscharakteristika ist. Wir untersuchen, ob und zu welchem Grad der Gesprächscharakter mit technischen Mitteln erfasst werden kann und legen dabei großen Wert auf die Betrachtung der Herausforderungen von mobilen Szenarien. Dafür prüfen wir, wie gut neueste Methoden mit den Auswirkungen von Hintergrundgeräuschen und Signalveränderungen umgehen, welche durch die Trageposition des Geräts verursacht werden. Wir stellen ein System vor, welches Gespräche des Nutzers erfasst und hinsichtlich in Sprache eingebetteter affektbezogener und sozialer Merkmale analysiert. Letztendlich zeigt sich, dass aus Audio gewonnene Informationen beider Art die durchschnittliche Genauigkeit von persönlichen Modellen der Benachrichtigungspräferenz jeweils um mehr als 20% verbessern (relativ zur Baseline der gängigen Kontextattribute).

Das diesem Bericht zugrundeliegende Vorhaben wurde mit Mitteln des Bundesministeriums für Bildung und Forschung unter dem Förderkennzeichen 01IS12057 gefördert. Die Verantwortung für den Inhalt dieser Veröffentlichung liegt beim Autor.

Acknowledgments

Multiple students have contributed significantly to this thesis, all of whom I'd like to thank for their hard work. In chronological order (submission date of thesis):

Martin Wieczorek (*A Black-Box Approach to Interruption-Free Notifications*, Bachelor's thesis, 2013)
Tobias Seitle (*Audio-Based Social Interaction Detection for Mobile Interruption Management*, Master's thesis, 2014)
Jitin Kumar Baghel (*Audio Based Characterization of Conversations*, Master's thesis, 2015)
Wahib Wahib-Ul-Haq (*Speaker Detection and Conversation Analysis on Mobile Devices*, Master's thesis, 2015)
Aiham Taleb (*Audio Based Characterization of Conversations with Channel Effect Compensation*, Master's thesis, 2015)
Hanna Schäfer (*Deriving Conversational Social Contexts from Audio-Data*, Master's thesis, 2015)
Zoran Ristevski (*Audio Contexts in Mobile Notification Management*, Bachelor's thesis, 2015)

Contents

List of Figures

List of Tables

1. Introduction and Outline

The greater part of the first section of this introduction has been published in [Schulze and Groh, 2014], further parts (cf. Chapter 13) are to appear in [Schulze and Groh, 2016]. All contents of this thesis constitute original contributions by the lead author.

1.1. Introduction and Motivation

Whenever a mobile device initiates interaction to push information, either because of incoming calls or messages or because of other proactive services, it runs the risk of interrupting the user in her current activity. The disruptive effects of interruptions on task performance have been well-studied and include slow recovery on task resumption, an increased number of mistakes, and emotional impairments like stress or frustration [Roda, 2011], [Mark et al., 2008].

The degree to which we are open to interruptions and insusceptible to their negative effects is referred to as *interruptibility* or *receptivity to interruptions* [Fischer et al., 2010]. It is subject to context and also interrelates with the type of interruption [Hansson et al., 2001]. Contextual factors can often be inferred and, in turn, allow inference of interruptibility. Alignment of notifications with interruptibility can be achieved, for example, by varying the timing of notifications or their modality (e.g. ringing, vibrating etc.).

Not only the user might be interrupted by a mobile notification but also her social environment. Social norms and our willingness to conform to them let social interruptibility be reflected in personal interruptibility: when we are aware of the interruptive effects of a notification on the people around us, we are likely to be less receptive to the interruption ourselves.

It is our general research goal to detect interruptibility in social situations in view of effectively managing mobile notifications. In particular, we look at situations with explicit, verbal interaction: conversations with user engagement.

While previous work has considered conversations in the context of interruption management, it didn't do so exhaustively: in most cases, only the presence of speech was taken into account

as a binary factor for determining interruptibility. This, however, doesn't capture only conversations in which the user engages as it doesn't distinguish personal conversations from background chatter. Very rarely and insufficiently was the speaker to whom the voice belonged taken into consideration. Furthermore, to the knowledge of the author, there is no study of how the character of a conversation affects the interruptibility of the involved speakers. While we can't reasonably argue that it is the sole essential factor for determining interruptibility, it is evident that the character of a conversation can play an important role. Our receptiveness to notifications in a heated argument may be different to that in the same argument, but in a relaxed atmosphere. Likewise, during a focused and goal-oriented discussion we're intuitively less interruptible than in a casual chat with a friend. Motivated by this intuition, we hypothesize that there are underlying factors in conversations that dictate an associated, characteristic receptiveness – an information that could by harnessed by mobile notification management systems for the benefit of the user.

For reasons of privacy, we restrict our research to information that is directly available to the user's personal mobile device. This contrasts information sharing strategies among multiple users.

1.2. Research Questions

In this body of work we seek to answer the following research questions:

RQ1a Do different types of conversations entail a characteristic interruptibility for individuals?

RQ1b If so, do characteristic interruptibilities translate across individuals?

RQ2 How well can conversational characteristics be inferred from audio from a technical standpoint?

RQ3a Is information about conversational characteristics useful for notification management?

RQ3b If so, what kind of characterization provides how much benefit?

RQ4 Is direct assessment of interruptibility based on low-level audio features more profitable than a two-step approach with an intermediate classification of the character of a conversation, which allows to incorporate additional socio-psychological domain knowledge?

1.3. Contributions and findings

In answer to our research questions, the following accomplishments and findings constitute our contribution:

1. We show that different kinds of conversation, defined according to various aspects, all have characteristic effects on people's receptiveness to interruptions. These effects translate across individuals. Some types of conversation exhibit an even higher associated receptiveness than situations without any verbal social interaction. This finding renders current practice of using the occurrence of a conversation as a binary indicator for unavailability inapt.
2. We reveal the richness of the acoustic medium in terms of social and affective information in the context of notification management and demonstrate that conversational characteristics derived from audio improve the accuracy of personal notification preference models on average by more than 20 percent. Social and affective information encoded in speech have similar utility, likely due to a large overlap in relevant content.
3. We find that in terms of classification performance, the explicit characterization of a conversation as an intermediate step provides no significant gain compared to implicit characterization in terms of low-level and medium-level audio features. However, we also point out and empirically back up the importance of an explicit characterization for giving the user the means to formulate rules in notification management.
4. We conduct the first extensive evaluation of *channel effects* (signal alterations) induced by smartphones as recording devices in everyday situations. We throughly compare positional variance and handset/microphone variance as the two main factors causing channel effects and find that the manner in which the user carries the smartphone clearly has the dominating impact. Furthermore, we show that
    a) there are no archetypal positions that allow for positional profiles as minimal changes in position or environment (clothing fabric) have great effect on the signal, and that
    b) conditioning on a multitude of different channel effects during training of the classifiers can help immensely to mitigate the impact on audio classification performance.

1.4. Outline

Part I: Background

Chapter 2 This thesis will open with a definition of the subject matter around which our research revolves: mobile notifications. We will show how notifications create interruptions which in turn often disrupt the user with severe consequences. Context-aware notification management will be introduced as a possible remedy. The state of the art in sensing, inferring, and integrating context on mobile devices will be presented, and we will show how currently captured attributes can be complemented by conversational context information to better cover the user's context in its entirety and to improve notification management.

Part II: Conversational Factors in Receptivity

Chapter 3 The fundamentals of speech and conversation will be established as a basis for conversation analysis.

Chapter 4 We will then elaborate on emotions and other affective phenomena as one potential aspect under which conversations can be characterized.

Chapter 5 A social taxonomy of speech events by [Goldsmith and Baxter, 1996] will be presented as basis for an alternative/complementing approach to classifying conversations.

Part III: Implementation

Chapter 6 After covering the conceptual part of our research, we will proceed to the implementation of a system for conversation analysis. For that, we will first introduce a set of essential mathematical tools.

Chapter 7 To identify suitable methods for analysis, a dataset that properly reflects our use-case scenario is needed. We will describe the methodology for synthesizing data that accurately simulates conversations encountered in real-life scenarios with many different kinds of background noise. As any conversation in our focus has to be captured by a mobile phone, we will put emphasis on studying so-called *channel effects*, i.e. signal alterations induced by the characteristics of the smartphone and the manner in which it is carried.

Chapter 8 A short overview of general audio signal processing and our pipeline will be presented.

Chapter 9 As the first of three research areas in speech processing on which we will touch, Voice Activity Detection is covered. We will identify methods suitable for separating speech from background noise and silence in our dataset. Although the better part of all evaluated algorithms performs similarly well, Random Forests will end up as our method of choice.

Chapter 10 After that, we will compare methods for identifying sections of speech that belong to a target user, in our case the owner of the smartphone. In accordance with current research, so-called *i-vectors* will turn out to be the most effective one under the constraint of channel effects.

Chapter 11 Emotion Recognition and Affect Regression will constitute the final area of research that we cover for conversation analysis. Despite the constraints imposed by our scenario (reduced feature set, noise, channel effects) we achieve results that don't deviate tremendously from results reported in recent literature. In absolute terms, however, they are only moderate. Best results in characterizing conversations with respect to their affective components are achieved with Long Short-Term Memory Recurrent Neural Networks.

Chapter 12 Eventually, we will present a system that integrates the above speech processing methods into a real-time system for analyzing conversations captured by the smartphone.

Part IV: Evaluation

Chapter 13 To evaluate our system and to study the utility of conversational context in mobile notification management, we will conduct a large-scale two-week user study with 54 users. This chapter constitutes the centerpiece of our research and most findings will be presented and discussed here.

Chapter 14 We will conclude this thesis with a summary of our efforts and an outlook on potential future research.

Part I.

Background

2. Interruptions – the Downside of Mobile Notifications

2.1. Mobile Notifications

We define the term *notification* as any cue emitted proactively by a mobile device with the goal of creating awareness of an underlying event which, according to the system, requires the user's attention, e.g. the arrival of a message or an incoming call. Cues can be acoustic, visual, tactile, or any other attribute that makes the device's new state distinguishable from the default, idle state.

Of course, different cues entail a different characteristic invasiveness. A flashing LED is by most definitions less invasive than loud ringing [Kern and Schiele, 2003, p. 3]. Invasiveness evidently has a strong effect on the time and manner the user's attention is obtained. Regardless of a notification's invasiveness, at the time the user notices the device's request for attention, she is interrupted in her current mental focus. The degree of this interruption determines the user's perception of it and the consequences. In the following sections we shed light on how interruptions are processed by the human mind and, subsequently, what negative effects they may cause.

2.2. Cognitive Processing of Interruptions

There are multiple ways how a user responds to a notification. [Latorella, 1999] proposes a model that formalizes the mental processing of a notification and the notification's effect on an ongoing task, in the literature often referred to as *primary task*. The model comprises four stages [Latorella, 1999, p. 21ff]:

> **Detection** The user is "engaged in an ongoing procedure" prior to the arrival of the notification. "[Her] activated memory contains representations associated with the ongoing procedure, and, in particular, those associated with the current task. [An] annunciation stimulus [(the notification)] heralds the interruption. If this stimulus is salient

enough to overcome sensory thresholds, it is stored in short-term sensory stores for further processing."

Interpretation If the stimulus is successfully detected, it is "translated to a working memory representation of the interrupting task in terms of its performance requirements". The translation is defined as *interpretation*. "[The user's] working memory now supports both representations associated with the ongoing procedure, specifically the interrupted task, and the interruption."

Integration If the notification is correctly interpreted with regard to the requirements the interrupting task entails, one of three actions is taken: the user either decides to schedule the interrupting task, to dismiss it or to perform it immediately. Either way, preemption of the ongoing task is required for the user to weigh "benefits associated with performing the interruption against costs of continuing the interrupted task".

Task resumption Eventually, the user continues performance of the original, interrupted task.

For an *in-depth* analysis of the cognitive processing of interruptions that is aligned with diverse theoretical models of the human memory we refer the reader to [Roda, 2011] and [Grundgeiger et al., 2010].

Literature in psychology that investigates interruptions and interruption management often mentions the terms *diversion, distraction, disturbance*, and *disruption* – sometimes interchangeably. In this thesis we focus on interruptions, the forced mental processing of a perceptually detected notification, and disruptions, the negative effects of processing the notification.

[Coraggio, 1990, p. 19] defines an interruption as "an externally-generated, randomly occurring, discrete event that breaks continuity of cognitive focus on a primary task." This highlights three important characteristics:

- The nature of the interruption is not controlled by the subject.
- The timing of the interruption is not known to the subject a priori. Coraggio clarifies: "This does not inply that [the subject] expects to be free of interruption. [The subject] may know that an interruption is possible – without knowing when (or if) it will happen."
- The interruption is of finite duration which distinguishes it from concurrent distractions like background noise.

While lasting distractions, too, can cause attentional conflicts that impair task performance [Groff et al., 1983], it is the disruptive nature of interruptions that can have tremendous negative impact on the user's tasks at hand [Speier et al., 2003]. This terminology is consistent with Latorella's definition of disruptions as "deleterious effects" [Latorella, 1999, p. 23] caused by the cognitive processing of an interruption. Latorella further elaborates on the reasons for these negative effects:

"*Interpretation requires attention resources to retrieve, or activate, long-term memory representations of the interrupting task; requires representation in working memory; and requires attention resources to maintain this working-memory representation. Capacity limitations and differentiation of these resources may result in deleterious effects. [...] If the [user] integrates the interruption, progress on the ongoing procedure is disturbed. Integration imposes additional attention and working memory requirements associated with preemption and resumption of the interrupted position. The execution of interruption response plans, and the process of scheduling when the interruption will be performed require attention and working-memory.*" [Latorella, 1999, p. 23]

From this explanation it follows that not every interruption leads to the same degree of disruption. The more attention is drawn from performance of the primary task and directed to the interruption, the more extensive is the disruption. However, it is evident that any notification processed by the user generates disruption to some extent, even when the interrupting task is dismissed and the interruption ignored.

2.3. The Costs of Interruptions

In the previous section we defined disruptions as negative effects resulting from limitations of memory capacity in the human brain and attention resources required to process interruptions. This section will further investigate the implications of disruptions.

In accordance with the common subjective notion that interruptions impair performance there is substantial empirical evidence for negative effects of interruptions when resuming the interrupted task:

Slow recovery Since any interruption needs to be processed, time passes between preemption and resumption of the primary task. This is called the task resumption lag [Grundgeiger et al., 2010]. The context associated with the primary task needs to be recalled and attention needs to be shifted back to the primary goal until focus is restored to a previous level. Depending on several factors, the task resumption lag can be quite significant. Evidently, this results in a longer total time spent on performing the primary task as is shown, for example, by [Kreifeldt and McCarthy, 1981], [Gillie and Broadbent, 1989], [Bailey et al., 2000], [Bailey and Konstan, 2006], and [Cutrell et al., 2001]. Naturally, it is also possible that interrupted users *forget* context information of the primary task like the point where they left off when the interruption occurred [Cutrell et al., 2001].

Mistakes Due to the attentional draw, users who are interrupted tend to commit errors in resumed performance of the primary task [Kreifeldt and McCarthy, 1981], [Gillie and Broadbent, 1989], [Cellier and Eyrolle, 1992], [Latorella, 1999], [McFarlane and

Latorella, 2002], [Bailey and Konstan, 2006], [Grundgeiger et al., 2010].

Emotional impairments According to an empirical study conducted by [Mark et al., 2008, p. 1] people who are interrupted often compensate for the delay by working faster at the cost of "experiencing more stress, higher frustration, time pressure, and effort". Along with that, interrupted users also experience annoyance and even anxiety [Bailey and Konstan, 2006], [Adamczyk and Bailey, 2004].
Furthermore, interrupted tasks are perceived more difficult to be completed than non-interrupted tasks [Bailey et al., 2001].

[Bailey et al., 2000, p. 5] also show that the resumption lag is related to the memory load of a task at the time of interruption and therefore to nature and complexity of the primary task: "A subject spent between 5% and 40% longer on an interrupted than a non-interrupted task. Again, a task having a higher memory load at the time of interruption demonstrated a relative increase in task completion time greater than that for a task having a smaller memory load." This finding has been backed up by follow-up experiments [Bailey et al., 2001] and by other researchers, e.g. [Speier et al., 2003] and [Czerwinski et al., 2004].

In addition to this individual-centered, cognitive viewpoint, disruptions can also be examined under the aspect of social norms. In [Kern and Schiele, 2003], the term *social interruptibility* is coined as a complement to *personal interruptibility*, putting emphasis on the social expectations of “proper conduct”. In consequence, [Kern and Schiele, 2003] advocate the consideration of the user’s social environment and the selection of less socially invasive modalities when applicable. A corresponding study of subtle, public notification cues for mobile devices is presented in [Hansson et al., 2001].

2.4. Avoiding Disruption Through Context-Aware Notification Management

It is evident that the goal of creating awareness of events on mobile devices must be balanced with the goal of minimizing personal and social disruptions as described in the previous sections. For example, the notification can be deferred [Horvitz et al., 2005a], the information content can be reduced [Streefkerk et al., 2012], or the modality (ringing, vibrating, etc.) be varied [Lopez-tovar and Charalambous, 2015]. Any such strategy requires the incorporation of context factors (cf. [Dey, 2001] for definitions of *context*), most often by means of sensors, that allow for inference of the importance of the notification content, the user’s personal as well as the social surrounding’s *receptivity* to the notification, in the literature also called *interruptibility*. Both terms refer to the degree to which subjects are open to a given interruption. [Ho and Intille, 2005, p. 1] illustrate:

"*Consider an office worker sitting at a desk discussing a report with a supervisor. If the phone rings and it is a co-worker with updated information for the report, the office worker is likely to be receptive to the phone call. However, if the phone call is from a friend to discuss plans for the weekend, then the office worker is likely to be less receptive. On the other hand, the office worker might be receptive to the phone call from the friend if the phone displays the message visually instead of using the ring to signal the interruption. The visual notification is less likely to disrupt the flow of the current conversation, perhaps lowering the perceived burden of the interruption for both people in the room.*"

2.4.1. Interruptibility and Context

In a well-grounded evaluation of the literature [Ho and Intille, 2005, p. 2] identified 11 context factors that influence a person's interruptibility:

Factor	**Description**
Activity of the user	The activity the user was engaged in during the interruption
Utility of message	The importance of the message to the user
Emotional state of the user	The mindset of the user, the time of disruption, and the relationship the user has with the interrupting interface or device
Modality of interruption	The medium of delivery, or choice of interface
Frequency of interruption	The rate at which interruptions are occurring
Task efficiency rate	The time it takes to comprehend the interruption task and the expected length of the task
Authority level	The perceived control a user has over the interface or device
Previous and future activities	The tasks the user was previously involved in and might engage in during the future
Social engagement of the user	The users role in the current activity
Social expectation of group behavior	Activities and expected reaction to interruption of nearby people
History and likelihood of response	The type of pattern the user follows when an interruption occurs

Table 2.1.: *High-level context factors that influence* ***personal interruptibility***

It should be noted that in the above model, social interruptibility is assumed to be reflected in personal interruptibility, i.e. as the individual's willingness to conform to social norms. While this might not necessarily be true, i.e., a person might not care about or misinterpret her

surroundings and might simply not acknowledge the de-facto social interruptibility ([Kern and Schiele, 2006, p.140] found only a 65% correlation of estimated social interruptibility for various public scenes between users), it is debatable whether detecting this discrepancy would be of any use. After all, one can make a point that the device must always act in accordance with the *owner's* expectations and intentions. Thus, our understanding of interruptibility is an integrated one.

In recent years, sensor-based statistical models of interruptibility have become the focus of a broad research community, most notably since the publication of [Fogarty et al., 2004] and [Fogarty et al., 2005] which arguably laid part of the field's foundation. As interruptibility is often low in the work place, a lot of research has been conducted in this, rather static, context, e.g. [Fogarty et al., 2004], [Oliver et al., 2004], [Avrahami et al., 2007]. With the advent of modern, sensor-rich smartphones and their ubiquity, a wider, more multifaceted context could be studied, e.g. [Fisher and Simmons, 2011], [Rosenthal et al., 2011]. In Table 2.2 we provide an overview of context descriptors found in recent literature on interruption management. As some authors use different sensors to measure similar factors, we grouped the results of low-level sensor processing and analysis into medium-level categories. Each category covers one or more *high-level* context factors [Ho and Intille, 2005] as listed above.

Medium-level context descriptor	**Examples**
Whether the user is talking	[Fogarty et al., 2004], [Avrahami et al., 2007]
Immediacy of conversation (e.g. phone, face-to-face)	[Oliver et al., 2004], [Rosenthal et al., 2011]
Computer activities (e.g. application, keyboard activity)	[Fogarty et al., 2004], [Oliver et al., 2004]
Movement (motion sensors)	[Rosenthal et al., 2011], [Fogarty et al., 2004]
Social cues (e.g. angle of office door)	[Avrahami et al., 2007], [Fogarty et al., 2004]
Relation to sender of mobile message	[Fischer et al., 2010], [Rosenthal et al., 2011]
Rating/valence of content	[Fischer et al., 2010]
Location and scenery (GPS, audio)	[Kern and Schiele, 2006], [Nagel et al., 2004]
Time and calendar information	[Rosenthal et al., 2011]
Task and interaction structure	[Iqbal and Bailey, 2010], [Iqbal and Bailey, 2008]
Phone posture	[Fisher and Simmons, 2011]

Table 2.2.: *Context factors that influence personal interruptibility*

A closer look at how and what auditory context is mentioned in interruption-related literature reveals the limited exploitation of this seemingly information-rich medium. [Kern and Schiele, 2006] classified background noises with respect to auditory scenery as an indicator of the user's (social) activity. Conversations were, however, not considered. So far, only the probability of a conversation and its immediacy have been taken into account in other publications, e.g. in [Fogarty et al., 2004], [Siewiorek et al., 2003] and [Oliver et al., 2004]. For the former, evidence was sometimes questionable: [Hashimoto et al., 2013] employed a Wizard of Oz approach and detection of any human voice in the past 20 seconds served as an indicator for the occurrence of a conversation. While that approach may work in very static contexts, it is not reliable in dynamic everyday environments that involve other people. [Sawhney and Schmandt, 2000] tackled this problem by first classifying audio segments to detect the voice of the target speaker. If detected, the user was categorically marked as non-interruptible. While speech detection and, to some degree, conversation-awareness have been harnessed for interruption management, no exhaustive research has been conducted on how the character of a conversation influences receptivity to mobile notifications, despite the variety of context factors (cf. Table 2.1) involved: evidently, conversation conveys information on the *activity of the user*, the *emotional state of the user*, the *social engagement of the user*, and the *social expectation of group behavior* (activities and expected reaction to interruption of nearby people). This renders conversations an ideal target for thorough study in the context of interruption management.

There are obvious limitations as to how well the context factors and, in turn, personal and social interruptibility can be inferred. As [Avrahami et al., 2007] demonstrate, even human prediction of another person's receptivity is highly prone to error. Cues are often missed or misinterpreted. Of course, analogous limitations apply to sensor-based models. However, it can be stated that overall they perform reasonably well in interruption management, often better than their human counterparts in various experiments. "[People] viewing [...] audio and video recordings can distinguish between 'Highly Non-interruptible' situations and other situations with an accuracy of 76.9%. A model based on [handful of very simple sensors] makes this same distinction with an accuracy of 82.4%. Both of these accuracies are relative to a chance accuracy of 68% that could be obtained by always estimating that a situation was not 'Highly Non-interruptible'." [Fogarty et al., 2004, p. 122]

2.4.2. Context integration

Knowing about the user's context and her presumed interruptibility is by itself not a remedy against disruptive notifications. The system has to incorporate contextual information in a suitable way into the decision-making progress of when and how to notify the user. Queuing

notifications and probing for moments with low interruptibility to deliver them in batches is an unsound *general* strategy for multiple reasons. Some notifications require immediate attention as their underlying event is urgent, e.g. an important phone call or the passing of a point of interest by a traveling user, where a belated notification is likely worse than an interruption. Also, the mental processing of a big set of queued notifications is monotonous and cumbersome. Some notifications might get dismissed without the user being aware of the corresponding event. Having this in mind, the evident goal for an intelligent notification system must be to *balance* awareness for events and receptivity to notifications [Horvitz et al., 2005a]. Strategies for this decision-making step can be grouped into three categories:

Utiliy based The most obvious approach is to formulate the balancing problem in quantifiable terms of utility and costs. This has been favored by many of the earlier publications by a group around Eric Horvitz, e.g. [Horvitz et al., 2002], [Horvitz and Apacible, 2003], [Horvitz et al., 2005b], [Kapoor et al., 2007], [Kapoor and Horvitz, 2008], but also others like e.g. [Chen and Black, 2008]. The term *expected cost of interruption* (ECI) was coined in [Horvitz et al., 1999] by means of the expected attentional status of the user (based on sensor evidence) and status-specific costs for a set of notification modalities. Costs had been assigned to combinations of statuses and notification modalities a-priori. The utility was defined in a relatively complex manner in terms of negative *expected costs for delayed action*, based e.g. on inferred urgency of a message. While the decision-theoretic approach by Horvitz allows for a nice mathematical formulation of the problem, it constitutes merely a very simplistic framework that covers neither the optimal assignment of the various costs (and thus utility) nor the inference of the user's attentional focus/state.

Machine learning based A more integrated, yet nontransparent way is the use of supervised machine learning techniques like classification or regression. Given a set of examples (training data), an optimal *direct* mapping of context data to notification modality or ringer volume is learned, e.g. in [Rosenthal et al., 2011] or [Ho and Intille, 2005]. Classifiers can also be trained to predict an optimal time for delivery of the notification or a reevaluation. While the decision-making process is nontransparent to the user, the acquisition of training data is much more natural and graspable as the user is only asked for how she wants her device to behave – although the user may be prompted frequently. This is opposed to the elicitation of rather cryptic numeric costs for *all* modalities, not just the desired one.

Rule based Entirely rule based systems mark a third approach. Here, complete control is given to the user who declares a set of rules by hand, e.g.: *If at the office, put the phone on vibrate unless call from parents.* This approach is not strictly orthogonal to the ones above and can be combined with either one, provided that rules can be phrased. That is often not the case with machine learning based "black box" systems.

In the more narrow context of *mobile* notification management, the machine learning approach has been reported to achieve satisfactory results, e.g. by [Rosenthal et al., 2011]. For the latter publication, a two-week user study was conducted for data acquisition and sensor readings were mapped directly to the desired volume of the smartphone's ringer. Averaged over the various groups the reported accuracy reached around 85%. Within the scope of a bachelor's thesis and associated experiments [Wieczorek, 2013], we were able to reproduce similar results, given enough training data ($>$ 40 samples). It must be noted that, according to the literature, e.g. [Rosenthal et al., 2011, p. 10], and our own experiments in [Wieczorek, 2013], there seem to be few universals in context-aware notification management. Individual preferences diverge strongly. Consequently, interpersonal notification preference models, i.e. models which are not trained from the target individual's data but from data of a plethora of other, supposedly representative individuals, perform far worse than personalized ones (56% vs 78% accuracy in [Wieczorek, 2013, p. 39]).

2.5. Summary

Notifications are a necessary means of proactive services and communication systems for creating awareness of an underlying event which requires the user's attention. In doing so, notifications interrupt the user in her current mental focus. Depending on that focus and the user's immersion (*interruptibility* or *receptivity*) and also on the invasiveness of the notification, the interruption may have grave, disruptive effects. To mitigate this problem, an intelligent system that manages the delivery of notifications and incorporates the user's current context can be employed. Various context factors can be inferred by means of sensors, integrated, and balanced to find an optimal time and modality for the notification.

Although auditory context has been considered in a rudimentary fashion, the richness of the medium has not been fully exploited. To the knowledge of the author, no closer evaluation of the social and affective information encoded in speech and their utility in interruption and notification management has ever been carried out. In this thesis, we aim at closing the gap by studying the use of conversation analysis for intelligent notification management.

Part II.

Conversational Factors in Receptivity

3. Fundamentals of Speech and Conversation

In order to conceptualize a system for conversation analysis it is important to first establish some fundamental linguistic terminology that will be used throughout this thesis. Foremost, basic phonetic units for the segmentation of speech have to be defined. [Laver, 1994, p. 110] distinguishes between feature, segment, syllable, (setting,) utterance, and speaking-turn (from smallest to largest).

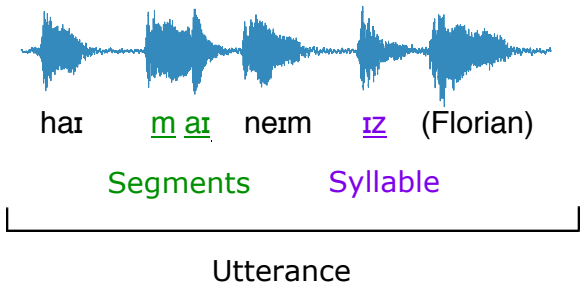

Figure 3.1.: *Phonetic units in a short sample utterance.*

The set of phonetic *features* constitutes the minimum set of descriptors used in order to account for the differences between phonetic units. The portion of speech with relatively constant phonetic features is called a phonetic *segment* (typically 30 to 200ms in duration [Fletcher and McVeig, 1992, p. 30ff.]). In audio analysis a feature (not to be confused with the phonetic feature) may be limited to a particular segment but may also be longer (*suprasegmental* or ***prosodic***, e.g. pitch) or shorter (*subsegmental* or ***spectral***). Segments are usually phonological units of the language, such as vowels and consonants. Multiple segments form a *syllable* which can also be defined in phonological terms and itself represents a level of higher linguistic organization. Finally, an *utterance* can be defined as a stretch of speech by a single speaker delimited by silence and containing no internal pauses whereas the *speaking-turn* consists of one or more utterances and denotes one speaker's contribution to a conversation. [Laver, 1994, p. 110 ff.], [Bernsen et al., 2012, p. 49], [Aronoff and Rees-Miller, 2003] "[M]ulti-utterance turns are the norm in natural dialogues" [Atterer et al., 2008, p. 1]. However, the number

of utterances per turn is highly dependent on context. [Panunzi et al., 2012, p. 160] reports an average number of about 1.5 utterances per turn for conversations/dialogues and 3.7 for monologues. These numbers are backed up by [Esposito et al., 2008, p. 110ff.] whose reports also indicate high variance in natural conversations.

Not every turn transition happens with an intermediate pause. Overlaps are frequent [Ten Bosch et al., 2005, p. 84ff.]. Also, not every transition attempt is successful. Sometimes, the second speaker tries to take over and interferes with a short utterance but the first speaker keeps the turn. [Weilhammer and Rabold, 2003, p. 2] therefore differentiate between five types of (attempted) turn transitions with different characteristics. Various speech parameters like for example speaking rate or the duration of pauses as the delimiter of both utterances and turns are subject to conversational context, speakers (sex, age, education, culture etc.), and language (a.o.). [Weilhammer and Rabold, 2003], [Jiahong et al., 2006], [Campione and Véronis, 2002a], [Kendall, 2013], [Grothendieck et al., 2009] A comprehensive study of universals and cultural variation in turn taking is presented in [Stivers et al., 2009].

There lies great informational value in the dynamics of conversational turn taking. For example, [Esposito et al., 2008, p. 119] elaborates on the reflection of situational dominance in turn taking patterns: "[Contrary] to naive assumptions of dialogue as a tennis-like exchange of question and answer or topic and comment, it actually presents a complex pattern of simultaneous talking as partners take turns to dominate in the interaction."

4. Affective Characterization of Conversations

In Chapter 1 we have motivated the characterization of conversations according to emotional aspects. In this chapter, we first clarify that emotions are only a subset of affective phenomena which we want to consider in their entirety. This allows us to also capture and interpret social cues embedded in speech. An overview of affective behaviors which manifest themselves in social cues is presented along with recent advances in computerized inference.

4.1. Fundamentals of Emotion and Affect

In psychology and cognitive sciences, the terms emotion, affect, feeling, and mood are frequently encountered and sometimes used interchangeably. According to [Ekkekakis, 2012, p. 1], they all constitute *affective phenomena*. [Shouse, 2005] elaborates: A feeling "is a sensation that is interpreted and labeled according to previous personal experiences" while emotion is the projection of that feeling, its display. "Feelings are personal and biographical, emotions are social, and affects are prepersonal. [...] Of the three central terms [...] feeling, emotion, and affect affect is the most abstract [...]." [ibd.] While emotions as projections can be captured computationally to some extent, affect can only be inferred. (Core) affect is a neurophysiological state that underlies simple feelings [Russell, 2009, p. 1] and is experienced constantly, although its nature and intensity vary over time [Ekkekakis, 2012]. Often, it is used as an "[...] umbrella term that covers all evaluative or valenced (i.e., positive/negative) states such as emotion, mood, and preference." [Juslin and Västfjäll, 2008, p. 561]. In this thesis, we will follow and emphasize this comprehensive interpretation by focusing on core affect and all emanating affective phenomena.

Recent developments in sensor technology gave rise to the research domain of *Social Signal Processing* (SSP) [Vinciarelli et al., 2009] in which this thesis is partly rooted. We focus on the affective components of social signals, i.e. of observable cues in interpersonal interaction that have (hidden) social meaning (see Section 4.2 below). For further information and different interpretations of fundamental terminology and concepts we refer the reader to the holistic

theoretical review in [Calvo and D'Mello, 2010].

4.1.1. Models and representations

Among many (e.g. in [Plutchik, 1991], [Ekman, 2005]), two dominant theoretical perspectives on emotions have emerged in affective computing [Hudlicka and Gunes, 2012], [Grandjean et al., 2008]. Both compete as general frameworks for characterizing emotions.

Categorical/discrete The arguably most intuitive and thus long-standing way of describing emotions (and other affective phenomena) is labeling them as discrete classes. Commonly, prototypical/basic emotion categories like happiness, sadness, fear, anger, etc. are used [Zeng et al., 2009, p. 39]. Most of the publicly available corpora for emotion recognition are labeled discretely, in turn, forcing subsequent publications to employ the same framework.

Dimensional/continuous However, a small set of basic emotions may often fail to capture affective nuances as "people exhibit non-basic, subtle, and rather complex affective states like thinking, embarrassment or depression" [Gunes et al., 2011, p. 827] . Thus, many researchers promote the use of dimensional models for underlying affective components. An early and widely used [ibd.] model is the *Circumplex Model of Affect* by [Russell, 1980] which is based on the two dimensions *valence* and *arousal*. In recent years, this model was extended by [Fontaine et al., 2007] with the two complementing dimensions *power* (also: control, potency) and *novelty* (also: predictability, expectation). While more complex, the dimensional approach has greater modeling power. Coordinates in the affective space can of course be mapped to discrete emotions.

4.1.2. Temporal Characteristics

To clarify what actually constitutes our object of interest with respect to time, we refer to [Cowie et al., 2001]'s temporal model of affect (cf. Figure 4.1). "Emotion in the narrow sense – full-blown emotions – are generally short lived and intense." [Cowie et al., 2001, p. 40] However, as audio is our sole source for inferring affective information, we cannot necessarily distinguish between temporal categories. If the user's voice reflects happiness, it may be due to short, recent event (→ emotion) or due to a general attitude over a longer stretch of time (→ mood). The distinction, although possibly relevant, requires the observation of the user's affective state over a long period (months) which is outside the scope of this work. We focus on the user's *predominant* affective state for a time span of several minutes to assess her punctual receptivity to notifications, ignoring the underlying cause for that state. We assume that this is a suitable period for capturing social interactions, more precisely: conversations, as

well as changes in the affective state caused by important events. These sudden changes are assumed to be most relevant for interruptibility. Thus, our temporal window of consideration will henceforth be referred to as *socio-emotional real-time*.

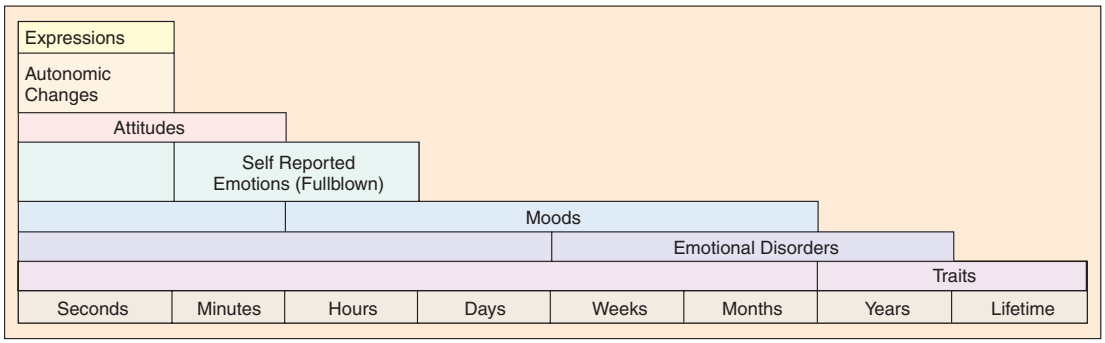

Figure 4.1.: *Temporal characteristics of affective categories [Cowie et al., 2001]*

4.2. Inference of Affective Phenomena from Audio

Nonverbal communication, a special case of human-to-human communication where the means used to exchange information consists of nonverbal cues [Richmond et al., 2008] in [Gökçay and Yildirim, 2011, p. 135], is of particular interest for our purposes as the cues are often displayed outside conscious awareness and can therefore be considered *honest signals* [Pentland, 2008]. Their social character is emphasized by the alternative term *social signal* [Vinciarelli et al., 2009]. [Gökçay and Yildirim, 2011, p. 135] elaborate on the close relation of affect and social signals, and state that the latter "leak reliable information about the actual inner state and feelings of people, whether they correspond to emotional state like anger, fear and surprise, general conditions like arousal, calm, and tiredness, or [affective behavior in form of] attitudes towards others like empathy, interest, dominance, and disappointment." All affective aspects of social signals can be subsumed under the umbrella term *affective phenomena*. Again, dimensional frameworks allow us to model the underlying affective components.

Evidently, social signals can have many forms: gesture, posture, gaze, et cetera [Vinciarelli et al., 2009]. In this body of work, we focus entirely on vocal signals that accompany verbal communication.

Several groups of features on multiple levels of granularity can be derived from the acoustics of speech. On the lowest level, spectral and prosodic features (cf. Chapter 3) like MFCC, PLP, pitch, energy, and speaking rate (tempo) can be found [Vinciarelli et al., 2009, p. 1747], [Gökçay and Yildirim, 2011, p. 144]. "The prosody conveys a wide spectrum of socially

relevant cues: emotions like anger or fear are often accompanied by energy bursts in voice (shouts) [...], pitch influences the perception of dominance and extroversion [...], the speaking fluency [...] increases the perception of competence and results into higher persuasiveness." [Vinciarelli et al., 2009, p. 1747] Silence, as a very basic, low-level element, is an important aspect of human communication and can convey many messages. As silence usually segregates stretches of speech, it is mostly considered in terms of pauses during verbal interaction that can reflect hesitation, difficulty in encoding or decoding speech, or consciously and openly displayed intentions with respect to the social interaction taking place: silence that signifies respect for people we want to listen to or commands that respect from others, silence that two lovers share, silence that ignores the other person and signals disinterest. [Richmond et al., 2008, p. 102], [Vinciarelli et al., 2009, p. 1747] Evidently, interpretation of silence is highly subject to context and thus other features. The direct relation to turn taking is obvious as pauses typically separate turns and give structure while a lack of coordinating pauses between turns indicates interruptions, hence a joint consideration suggests itself. The relevance of turn taking with respect to roles in social interaction was already touched upon in Chapter 3. Turn transition characteristics, turn length, and overall turn share are strong indicators for (im)patience and, going further, (in)equalities in status and dominance. [Sacks et al., 1974, p. 700], [Smith-Lovin and Brody, 1989, p. 424, 430 ff.] Non-linguistic vocalizations (also: vocal outbursts) like laughing, crying, groaning etc. mark another important group of features, especially in regard to affect and social situations: "[Laughter] tends to reward desirable social [behavior ...] and shows affiliation efforts, while crying is often involved in *mirroring* [...], that is the mutual imitation of people connected by strong social bonds [...]. Also, research in psychology has shown that listeners tend to be more accurate in decoding some basic emotions as well as some non-basic affective and social signals such as distress, anxiety, boredom, and sexual interest from vocal outbursts like laughs, yawns, coughs, and sighs [...]." [Vinciarelli et al., 2009, p. 1747] While sometimes regarded as non-linguistic vocalizations, yelling and whispering are considered *modes* of speech in this thesis since they commonly contain linguistic content.

As depicted in Figure 4.2, inference of high-level affective phenomena may happen progressively along the levels of inferential granularity, i.e. with intermediate steps, or directly from low-level features.

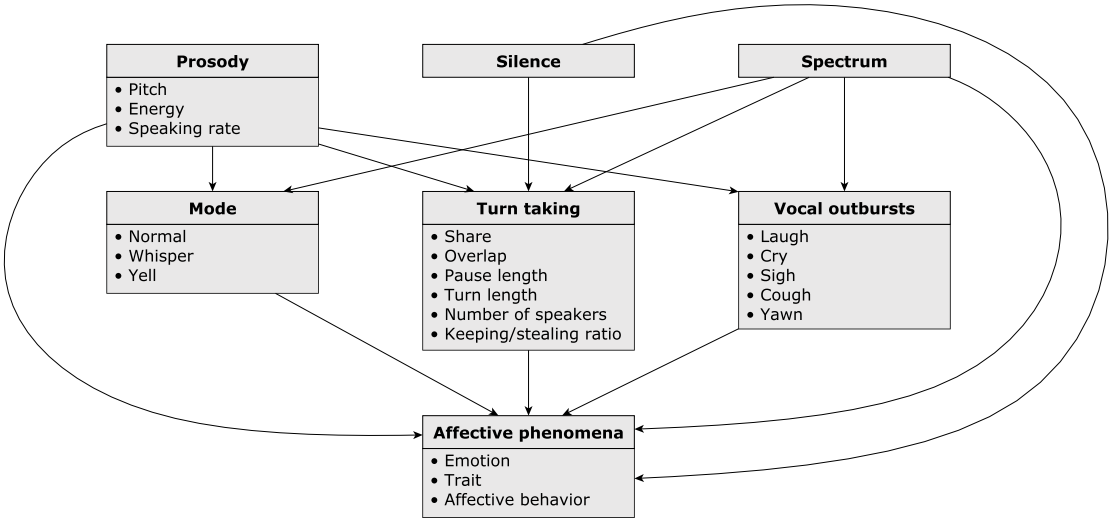

Figure 4.2.: *Feature groups derived from the acoustics of speech and their relevance for affect recognition (based on [Vinciarelli et al., 2009, p. 1747 ff.], [Sacks et al., 1974, p. 700], [Gökçay and Yildirim, 2011, p. 144 ff.] and own contributions)*

Recent literature, perhaps most notably [Madan and Pentland, 2006], lists many successful attempts in inferring particular affective phenomena from audio. Common affective phenomena successfully derived from low-level audio in recent research are (a.o.):

- Dominance/role/status
- Persuasion
- Interest
- Attraction
- Puzzlement
- Influence/engagement
- Emphasis
- Consistency
- Agreement/rapport
- Antagonism/rebelliousness
- Mimicry/mirroring
- Empathy
- Activity
- Solidarity
- Willingness to compromise

For more information and extensive surveys including performance results we refer the reader to [Gökçay and Yildirim, 2011, p. 145 ff.], [Zeng et al., 2009, p. 43], [Pentland, 2004], [Salamin et al., 2013] (also the sources for the above list).

5. Social Characterization of Conversations

In Chapter 4 we have proposed an affective characterization of conversations for providing conversational context information in notification management. In this chapter, we advocate the use of a second manner for characterizing conversations, one with a greater focus on interpersonal, social properties. For that purpose, we employ an already established taxonomy of conversations from the field of communication research. It must be noted that the two manners of characterizing conversations are not disjoint: an affective taxonomy/approach naturally comprehends social elements (cf. list of affective phenomena) just like a social taxonomy involves affect. However, we *hypothesize* that both manners of characterization have unique aspects and are not equivalent. Intuitively, affect is more individual-centric. We will revisit this point later on in this thesis.

5.1. An interpersonal taxonomy of speech events

In a series of three studies, [Goldsmith and Baxter, 1996] systematically developed "a descriptive taxonomy of jointly enacted speech events in everyday relating". In their first study, 903 open-ended diary log entries of 48 individuals monitoring their interactions over a course of four weeks were collected and subjected to systematic interpretative analysis in order to identify: "(a) types of talk respondents recognized, (b) commonly used labels for referring to types of talk, and (c) semantic dimensions used to differentiate among types of talk" (ibid.). A preliminary taxonomy of 24 speech events was constructed. In the second study, exhaustiveness of the taxonomy and generalizability w.r.t. age, sex, and ethnic differences were verified and optimized, resulting in a refined taxonomy of the following 29 events:

1. Gossip
2. Making Plans
3. Asking a Favor
4. Reminiscing
5. Joking Around
6. Asking Out
7. Making Up
8. Catching Up
9. Small Talk
10. Conflict

11. Serious Conversation
12. Talking About Problems
13. Love talk
14. Breaking Bad News
15. Recapping
16. Complaining
17. Sports Talk
18. Getting to Know
19. Group Discussion
20. Persuading
21. Decision Making
22. Getting/Giving Instructions
23. Class Information Talk
24. Lecture
25. Interrogation
26. Morning Talk
27. Bedtime Talk
28. Relationship Talk
29. Current Events Talk

In their third study, Goldsmith and Baxter systematically (in accordance with [Kruskal and Wish, 1978]) identified three dimensions using multiple regression analyses on conversational attributes collected in the previous studies:

> ***Dimension 1*** *indicates the extent to which a speech event is perceived as formal and goal-oriented.*
> ***Dimension 2*** *captures the extend to which a speech event is perceived as important, deep, and involved.*
> ***Dimension 3*** *refers to the positive valence of a speech event (although the attribute fit on this dimension is less robust than for the first two* dimensions)

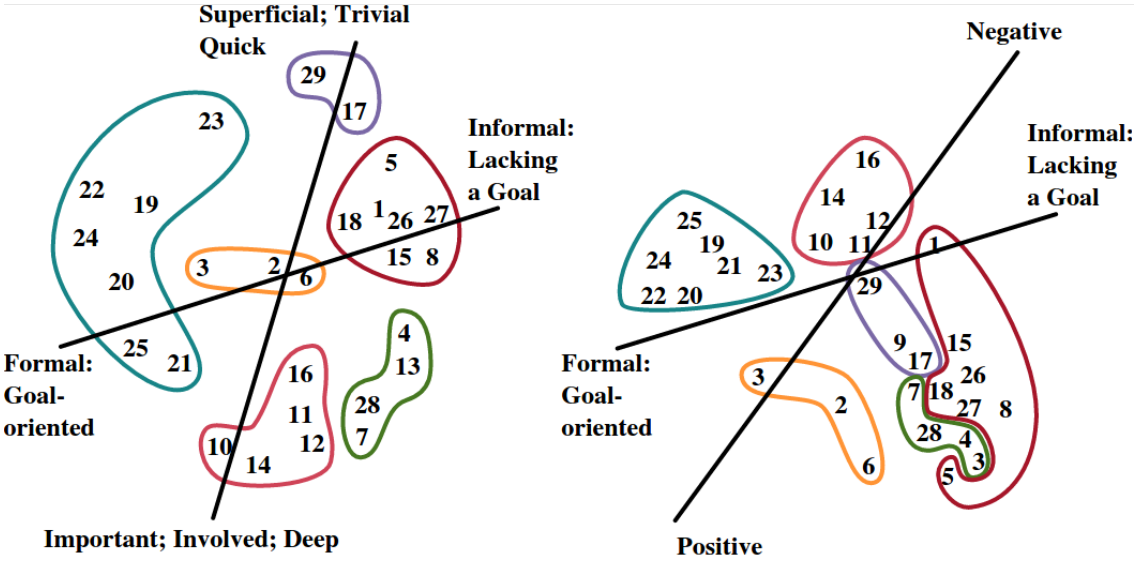

Figure 5.1.: *Two-dimensional plots of speech events and their clusters according to [Goldsmith and Baxter, 1996, p. 101]*

Figure 5.1 depicts two-dimensional plots of all 29 speech events along combinations of the above dimensions. By means of hierarchical cluster analysis Goldsmith and Baxter were also able to identify six clusters of speech events [Goldsmith and Baxter, 1996, p. 101], subsequently called *conversation types* (colors refer to figure):

- **Superficial talk**
- **Informal talk**
- **Involving, positive talk**
- **Negatively valenced talk**
- **Formal goal-directed talk**
- **Less formal goal-directed talk**

5.2. Linking the taxonomy to interruptibility

In Chapter 1 we motivated our research with concrete examples of conversational situations which intuitively entail a different receptivity to any kind of interruption: a heated argument, a casual chat, a focused discussion. We find our examples to be well reflected in both Goldsmith and Baxter's conversation types and conversational dimensions. The dimension of formality seems to capture the social aspect of interruptibility, while valence and depth might typify more of a personal component.

In a pre-study [Ristevski, 2015, p.32 ff.] we verified the adequacy of the six conversation types (and, thus, also the dimensions) as context source for determining a person's interruptibility. This pre-study was conducted in form of a survey among 57 participants in Munich, Germany. The sample included 2 teenagers (14-18y), 22 young adults (18-25y), 29 adults (25-65y), and 4 senior citizens (>65y). The survey can be found in Appendix A. It provided information about the context of our research and the relation to notification management. Every participant was asked to assign a personal reachability level ranging from 1 to 5 – 1 being the lowest reachability and 5 the highest – to each of the 29 speech events. The resulting scores were averaged over the six conversation types (clusters of speech events) resulting in 57×6 interruptibility scores. To see if these scores are statistically significant, we analyzed our within-subject study [Martin, 2004, p. 148] by means of a Friedman test [Greene and D'Oliveira, 1985, p. 62 ff.] (non-parametric counterpart of an ANOVA) ($\chi^2 = 175.4, p = 5.3\text{e-}36$) and follow-up multiple pairwise Wilcoxon signed-rank tests [Falk et al., 2014, p. 150] with Bonferroni-Holm correction [Holm, 1979]. All pairwise comparisons showed significance on the corrected .05 level except for *Involving, positive talk* vs *Less formal goal-directed talk* and *Negatively valenced talk* vs *Formal goal-directed talk*. Exact results are listed in Appendix B. Figure 5.2 shows a boxplot of mean interruptibility scores by conversation type.

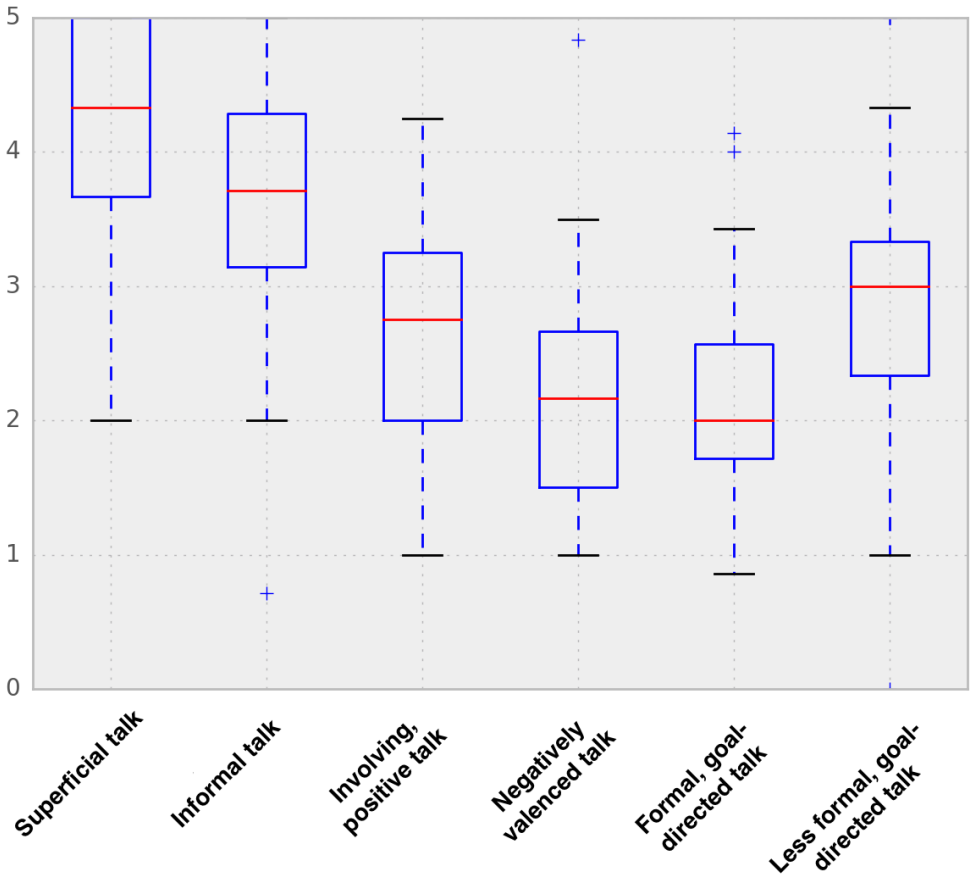

Figure 5.2.: *Survey results: mean interruptibility by conversation type (values seemingly cut off are exactly 5)*

The significant differences clearly indicate that a characterization of conversations with respect to social properties (according to [Goldsmith and Baxter, 1996]) provides a suitable and promising context source for determining a person's receptivity to notifications. The pre-study also serves the purpose of giving us a notion of the direction of differences between pairs of conversation types which allows us to use one-sided statistical tests with greater confidence in the evaluation of the final user study (cf. Chapter 13). This of course assumes that tendencies found in reported survey values still hold in real life. As we have no reason to believe that the population of both samples (pre-study and final study) are different, we make that assumption for those conversation types that already exhibited significant differences here.

Part III.

Implementation

6. Methods

This chapter gives definitions for all signal processing and machine learning methods used throughout the thesis for voice activity detection, speaker recognition, and affect recognition. As they are well-described in the literature we refer the reader to the respective sources for introductions and more detailed explanations. It must be noted that the features below mark a set for *experimentation*. The selection of features for the *final system* will be addressed in the according chapters.

6.1. Features

6.1.1. Spectral Entropy

The entropy of a signal's spectrum gives cues about the frequency pattern of the sound [Lu et al., 2009]. "In the frequency domain, voiced frames have [...] low spectral entropy. On the other hand, [...] unvoiced frames or non-speech frames [...] yield relatively high spectrum entropy." [Lu et al., 2011, p. 7] Following a Fourier transform of the signal, the probability density p_i of the spectrum, i.e., of all N frequency components, is computed as $p_i = s(f_i)/\sum_{k=1}^{N} s(f_k)$ [Shen et al., 1998, p. 1] where $s(f_i)$ is the spectral energy of frequency component i. Spectral entropy is then obtained through $H = -\sum_{i=1}^{N} p_i \log p_i$ [Shen et al., 1998, p. 2]. This feature is commonly used for *Voice Activity Detection*.

6.1.2. Pitch

Pitch is a perceptual property that is "related to the fundamental frequency of vibration of the vocal cords over some duration" [Beigi, 2011, p. 132]. The fundamental frequency, denoted f_0, is the lowest frequency of a periodic signal. Due to the close relation of pitch and fundamental frequency both terms are often used synonymously [Kinnunen, 2003, p. 30]. The pitch of the human voice lies in a characteristic frequency range [Furui, 2001, p. 250 ff.] which allows for the separation of speech from silence and, to some degree, even background noise [Xu et al., 2013], [Lu et al., 2009]. It varies between individuals and emotions,

hence it is commonly used for all speech-related tasks. To estimate f_0 we use the noise-robust PEFAC algorithm [Gonzalez and Brookes, 2014] which minimizes a cost function in the frequency spectrum [Gonzalez and Brookes, 2014, p. 523] to select the statistically most probable among several f_0 candidates (peaks).

6.1.3. Zero Crossing Rate

Zero Crossing Rate (ZCR) is defined as the number of zero-crossings of all sample values within a frame and can therefore be considered a lightweight approximation of the pitch [Lu et al., 2011, p. 5]. It can be computed step by step using

$$ZCR_f = \frac{\sum_{i=0}^{n} |\operatorname{sgn}(s_i) - \operatorname{sgn}(s_{i-1})|}{2} \tag{6.1}$$

where s_i denotes the i-th sample in an n-sample frame [Lu et al., 2009, p. 5]. This feature is commonly used for *Voice Activity Detection*.

6.1.4. Mel Frequency Cepstral Coefficients

At the time of writing (2014), Mel Frequency Cepstral Coefficients (MFCCs) constitute the most dominant feature in the Speaker Recognition literature. Although various other features can outperform MFCCs in specific settings, MFCCs are estimated as “difficult to beat in practice” [Kinnunen and Li, 2010].

MFCCs are a representation of the short-term power spectrum of an audio signal. They are computed with the aid of the "[psycho-acoustically] motivated [Mel] filterbank" [Kinnunen and Li, 2010, p. 5]. After the power spectrum of a windowed frame is obtained with the discrete Fourier transform, the powers are mapped onto the frequency-warped Mel scale $\mathscr{M}$ [Stevens et al., 1937, p. 187ff.] with M channels and their logarithm is computed. These last two steps are sometimes interchanged. In the final step a discrete cosine transform is performed on the log Mel powers yielding n Mel Frequency Cepstral Coefficients. Formally, we write [Beigi, 2011, p. 169ff.], [Kinnunen and Li, 2010]:

$$c_n = \sum_{m=1}^{M} \log(\mathscr{M}_m) \cos\left[\frac{\pi n}{M}(m - \frac{1}{2})\right], \quad n \in (0,)1, .., K. \tag{6.2}$$

$K \leq M$ is the number of coefficients one wants to obtain. It is common to exclude c_0 as it represents the average energy of the frame and doesn’t provide discriminative information

with respect to the speaker [Lu et al., 2011, p. 7].

In speaker recognition, it is common to perform Cepstral Mean and Variance Normalization (**CMVN**) [Togneri and Pullella, 2011, p. 29], [Kinnunen and Li, 2010], [Strand and Egeberg, 2004], [Westphal, 1997]. Although the term suggests otherwise, it is not to be confused with "global" centering and scaling to unit variance, a common practice in machine learning [Hsu et al., 2010, p. 4]. CMVN is a short-time normalization procedure over a running or fixed-length window. Usually, the goal of normalization is to remove bias towards features with bigger numerical values in distance-based optimization. CMVN, however, aims at removing noise and other non-discriminative information that is assumed to be static over the window but dynamic in general, e.g. channel effects or colored noise.

MFCCs represent the spectrum of the frame and are therefore static over the duration of the frame. To incorporate time-dynamic information in frame-wise processing we augment (concatenate) the feature vector of coefficients with temporal first-order derivatives ("deltas") [Bimbot et al., 2004, p. 433].

We perform MFCC extraction with VOICEBOX[1] for Matlab [The MathWorks Inc., 2014] 2014a.

6.1.5. (RASTA) Perceptual Linear Prediction

As an alternative to Mel frequency cepstral analysis, perceptual linear prediction (PLP) has emerged [Hönig et al., 2005]. Both are very similar in nature and are based on the short-term spectrum of speech. In PLP frequencies are warped according to the Bark Scale $\mathscr{B}$ [Zwicker, 1961]. “Differences between PLP and MFCC lie in the filter-banks, the equal-loudness pre-emphasis, [and] the intensity-to-loudness conversion” [Hönig et al., 2005, p. 2997]. For details we refer the reader to [Hermansky, 1990], [Cheng et al., 2005] and [Hönig et al., 2005].

To boost robustness of PLP coefficients, [Hermansky and Morgan, 1994] introduced RelAtive SpecTrAl filtering, RASTA for short. “[The human] ear is more sensitive to certain modulation frequencies, and the RASTA processing attempts to filter out unimportant modulation frequencies.’ [Kinnunen, 2003, p. 84]’ “Compared to CM[V]N, RASTA has an advantage in that [its application] to the matched clean case produces negligible decrease in performance.” [Togneri and Pullella, 2011, p. 40] “In general, it seems that conventional features like MFCC can outperform PLP in clean environment [sic.], but PLP gives better results in noisy and mis-matched conditions.” [Kinnunen, 2003, p. 76]

To extract PLP coefficients and to perform RASTA filtering we employ the RastaMat library[2].

[1] `http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html`, accessed July 9th, 2014
[2] `http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/`, accessed July 9th, 2014

6.2. Classifiers

6.2.1. Gaussian Mixture Model

A Gaussian Mixture Model (GMM) can be considered a superposition of multiple Gaussian distributions to estimate a single target distribution. In the multivariate case each Gaussian base distribution is characterized by its mean $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$. Individual weights π_k which sum up to 1 determine the influence on the joint mixture model. All model parameters are given by $\boldsymbol{\theta}$. Thus the GMM with K components (Gaussians) has the form

$$p(\mathbf{x}|\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k). \tag{6.3}$$

Multiple methods exist for the estimation of $\boldsymbol{\theta}$ [Görür and Rasmussen, 2010], [Herranz, 1999]. Throughout this thesis the Expectation-Maximization (EM) algorithm with randomized K-means initialization is employed. For a detailed description we refer the reader to [Bishop, 2006, p. 435ff.]. To alleviate computational complexity we compute only diagonal covariance matrices, e.g. $(\Sigma_k = diag(\sigma_1^2, \sigma_2^2, ...,)_k)$, which is also empirically known to improve results in speech processing [Togneri and Pullella, 2011, p. 31].

For general classification tasks each candidate class is represented by its own mixture model λ_c, i.e., by the class-conditional (speaker-dependent) density. For any sample vector $\mathbf{x}$ the class of the mixture with the highest posterior probability $\underset{c}{\operatorname{argmax}}\, p(y = c|\mathbf{x}, \boldsymbol{\theta})$ is chosen. Since all classes share the normalization factor the maximum a posteriori (MAP) estimate $\hat{y}_{MAP} = \underset{c}{\operatorname{argmax}}\, p(\mathbf{x}|y = c, \boldsymbol{\theta})p(y = c|\boldsymbol{\theta})$ is used – the latter factor being the class prior. Evidently, in a two-class scenario the decision function can also be formulated as a comparison of the log likelihood ratio of both classes with a threshold.

In each iteration of the basic EM algorithm $\Theta(NK)$ computations [Bishop, 2006, p. 439] are performed. For large UBMs and training sets empirical results on a 3.4Ghz Intel Xeon exhibited computation times of several weeks and longer. We therefore employ the highly parallel, GPU-based PyCASP [Gonina, 2013], at the time of writing the only CUDA-enabled GMM framework written in Python.

6.2.2. Gaussian Naive Bayes

Like the Gaussian Mixture Model Naive Bayes models a class-conditional distribution that can be used for a generative classifier. It assumes all features to be conditionally independent of one another given the class label. We formulate the class-conditional density for a

D-dimensional feature vector as a product of one-dimensional densities [Murphy, 2012, p. 84]:

$$p(\mathbf{x}_| y = c, \boldsymbol{\theta}) = \prod_{j=1}^{D} p(x_j | y = c, \boldsymbol{\theta}_{jc}) \tag{6.4}$$

In the case of real-valued features, individual univariate Gaussians can be used, yielding a Gaussian Naive Bayes (GNB):

$$p(\mathbf{x}|y = c, \boldsymbol{\theta}) = \prod_{j=1}^{D} \mathcal{N}(x_j | \mu_{jc}, \sigma_{jc}^2) \tag{6.5}$$

In speech processing many features are decorrelated (e.g. MFCC, PLP; see Section 6.1.4). From this it follows that with respect to our data the independence assumption of Naive Bayes holds.

Because of its simplicity, fitting the model can be accomplished in $\mathcal{O}(DN)$ time. We refer the reader to [Murphy, 2012, p. 85] for details. Our implementation is based on scikit-learn [Pedregosa et al., 2011] v0.15.

6.2.3. Logistic Regression

Being a binary discriminative classifier logistic regression directly estimates the posterior probability by means of the logistic sigmoid function [Murphy, 2012, p. 21]:

$$p(y = 1|\mathbf{x}, \mathbf{w}) = \text{sigm}(\mathbf{w}^\mathsf{T}\mathbf{x}) \quad \text{with} \tag{6.6}$$

$$\text{sigm}(\eta) \triangleq \frac{e^\eta}{e^\eta + 1}. \tag{6.7}$$

We fit the model using a trust region Newton method with L2 regularization as described in [Lin and Weng, 2008].

We employ scikit-learn [Pedregosa et al., 2011] v0.15 and liblinear [Fan et al., 208] v1.94.

6.2.4. Support Vector Machine

Support Vector Machines (SVM) for classification (SVC) are binary discriminative classifiers that try to separate two classes in a set of D-dimensional samples by means of an D-1-dimensional hyperplane with a maximized margin between the plane and the nearest data point of any class. Let $\mathbf{x}_i \in \mathbb{R}^D$ be a vector in a training set of size N and $\boldsymbol{y} \in \mathbb{R}^N$ be an indicator vector

such that $y_i \in \{-1, 1\}$. The dual form of this primal optimization problem can be formulated as

$$\min_{\boldsymbol{\alpha}} \frac{1}{2}\boldsymbol{\alpha}^\intercal Q\boldsymbol{\alpha} - \boldsymbol{e}^\intercal\boldsymbol{\alpha} \quad \text{subject to} \quad \boldsymbol{y}^\intercal\boldsymbol{\alpha} = 0 \tag{6.8}$$

$$\text{with} \quad 0 \leq \alpha_i \leq C, \quad \boldsymbol{e} = [1, ..., 1]^\intercal, \tag{6.9}$$

$$\text{and} \quad Q_{ij} \equiv \mathcal{K}(\mathbf{x}_i, \mathbf{x}_j), \tag{6.10}$$

which constitutes a so-called C-SVC [Chang and Lin, 2011, p. 3ff.]. The optimal value of C is determined empirically using cross-validation. An unknown vector $\mathbf{x}$ can then be classified using the decision function[3] $\text{sgn}(\sum_{i=1}^{N} y_i \alpha_i \mathcal{K}(\mathbf{x}, \mathbf{x}_i) + b)$. Extension of the concept to regression of continuous data can easily be achieved [Smola and Scholkopf, 2004].

We employ the well-known [Murphy, 2012, p. 482], [Bishop, 2006, p. 299ff] infinite-dimensional radial basis kernel $\mathcal{K}(\mathbf{x}, \mathbf{x}') = \exp(-\gamma|\mathbf{x} - \mathbf{x}'|^2)$ which depends on the parameter $\gamma > 0$ ($\frac{1}{N}$ by default [Scikit-learn, 2012, p. 618]).

Our implementation utilizes scikit-learn [Pedregosa et al., 2011] v0.15 and libsvm [Chang and Lin, 2011] v3.18. Note that the above optimization problem is a quadratic program so even sophisticated state-of-the-art solvers take between $\mathcal{O}(N^2)$ and $\mathcal{O}(N^3)$ time [Murphy, 2012, p. 501]. To speed up the model computation at the expense of precision we also try out stochastic gradient descent (SGD) which only approximates the gradient during optimization. The algorithm as well as the resulting SVM are described in [Tsuruoka et al., 2009]. With SGD we stop after 100 iterations.

6.2.5. Random Forest

Random Forests are ensembles of multiple decision trees with the decision function

$$f(\mathbf{x}) = \mathbb{E}[y|\mathbf{x}] = \sum_{m=1}^{M} \frac{1}{M} f_m(\mathbf{x}) \tag{6.11}$$

where the m-th tree is given by

$$f_m(\mathbf{x}) = \sum_{l=1}^{L} w_l \mathbb{I}(\mathbf{x} \in R_l) = \sum_{l=1}^{L} w_l \phi(\mathbf{x}; \mathbf{v}_l). \tag{6.12}$$

In the above formula $\mathbb{I}$ is the binary indicator function, R_l denotes the l-th region in space, w_l is the mean response in this region and $\mathbf{v}_l$ encodes the choice of variable to split on [Murphy,

[3] The computation of b is part of the model fitting procedure.

2012, p. 546, 553]. Individual trees are trained on random subsets of both the samples and the features. Further details on fitting the models can be found in [Breiman, 2001].

To measure the quality of a split we first compute the class-conditional probability

$$\hat{\pi}_c = \frac{1}{|\mathscr{D}|} \sum_{i \in \mathscr{D}} \mathbb{I}(y_i = c) \tag{6.13}$$

where $\mathscr{D}$ is the data in the respective leaf [Murphy, 2012, p. 549].

Given $\hat{\pi}_c$ we consider two split criteria in our experiments [Murphy, 2012, p. 549ff.]:

- Entropy: $-\sum_{c=1}^{C} \hat{\pi}_c \log \hat{\pi}_c$
- Gini impurity: $1 - \sum_{c=1}^{C} \hat{\pi}_c^2$

A split is discarded if after the split one of the leaves would contain less then $\delta \in \{1, \lg N, \ln N\}$ samples. This constraint is tree-specific. Both the split criterion and δ are determined empirically using cross-validation.

scikit-learn [Pedregosa et al., 2011] v0.15 provides us with a fast and cache-friendly implementation of the Random Forest classifier. For experiments with small numbers of samples (less memory consumption) we also use CudaTree[4] v0.6, a massively parallelized GPU-powered framework.

6.2.6. Long Short-Term Memory Recurrent Neural Network

Let g be a non-linear *activation* function and H a number in $\mathbb{N}$. Furthermore, let $\boldsymbol{T}$ be a 2nd order tensor consisting of two weight matrices $\boldsymbol{\Theta}^{(1)}$ and $\boldsymbol{\Theta}^{(2)}$. We define

$$\mathbf{z}(\mathbf{x}) = g(\boldsymbol{\Theta}^{(1)}\mathbf{x}) = [g(\boldsymbol{\theta}_1^{(1)\intercal}\mathbf{x}), ..., g(\boldsymbol{\theta}_H^{(1)\intercal}\mathbf{x})] \quad \text{and} \tag{6.14}$$

$$p(\mathbf{y}|\mathbf{x}, \boldsymbol{T}) = \text{Cat}(y|\mathscr{S}(\boldsymbol{\Theta}^{(2)}\mathbf{z}(\mathbf{x}))) \tag{6.15}$$

where $\mathscr{S}$ is the sum-to-one constraint and Cat denotes the multinoulli distribution [Murphy, 2012, p. 283, 308, 563ff.]. This constitutes a basic feedforward neural network. $\mathbf{z}$ is called the hidden layer with H hidden units. By recursively using the *output* $\mathbf{z}^{(i)}(\mathbf{x})$ of a hidden layer i as *input* for a subsequent layer $i+1$, i.e., $\mathbf{z}^{(i+1)}(\mathbf{z}^{(i)}(\mathbf{x}))$, we can construct a network with multiple hidden layers of varying sizes $H^{(i)}$ and with custom layer weights $\boldsymbol{\Theta}^{(i)}$.

If nodes in one layer are not only connected to nodes in the subsequent layer, i.e. there are connections within one layer or to a previous one, we speak of a recurrent network (RNN)

[4] `https://github.com/EasonLiao/CudaTree`, accessed July 1st, 2014

as opposed to a feedforward network. [Murphy, 2012, p. 570] This allows for the modeling of time-dependent data. However, only short temporal dependencies can be captured due to the so-called vanishing gradient problem in training (the backpropagated error blows up or decays exponentially [Eyben et al., 2009, p. 12]). The Long Short-Term Memory RNN, an architecture unaffected by this problem, was introduced by [Hochreiter and Schmidhuber, 1997]. It uses memory cells to store information over longer stretches of time. Illustrations and formal descriptions can be found in [Graves, 2013].

We experiment with two different GPU-accelerated implementations: a python-based implementation using Theano [Bastien et al., 2012] (optimizer: RMSProp [Dauphin et al., 2014], cost function: cross-entropy [Rubinstein and Kroese, 2013]) and a C++ implementation using Currennt [Weninger and Bergmann, 2014] (optimizer: standard gradient decent, cost function: SSE [Amaral et al., 2013]).

6.2.7. Factor Analysis

Factor analysis is a modeling method that can be used for generative classification, much like Gaussian mixture models. An observation $\mathbf{x} \in \mathbb{R}^D$ is assumed to be generated by a hidden class variable $\mathbf{z} \in \mathbb{R}^L$ so factor analysis is a latent model of the form [Murphy, 2012, p.383]

$$p(\mathbf{x}|\mathbf{z}, \boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{W}\mathbf{z} + \boldsymbol{\mu}, \boldsymbol{\Psi}). \tag{6.16}$$

In this equation $\boldsymbol{W} \in \mathbb{R}^{D \times L}$ denotes the so-called *factor loading matrix* whose columns span a linear subspace within the data space [Bishop, 2006, p. 571]. $\boldsymbol{\Psi} \in \mathbb{R}^{D \times D}$ is a diagonal covariance matrix. The underlying assumption of factor analysis is that the latent factors $\mathbf{z}$ "will reveal something interesting about the data" [Murphy, 2012, p.384].

To fit the model we employ Bob v. 1.2.2 [Anjos et al., 2012] which uses a derivation of the EM algorithm.

6.3. Auxiliary Transformations

6.3.1. Whitening

The whitening transformation, also called sphereing, is a decorrelation technique that ensures the data have a covariance matrix equal to the identity matrix $\mathbf{I}$ and equal variance along each dimension. As [Murphy, 2012, p. 144] explains, this can be done by computing $\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}^\intercal\mathbf{x}$ for each vector $\mathbf{x}$, with $\mathbf{U}$ being the eigenvectors and $\mathbf{\Lambda}$ the eigenvalues of the mean-free covariance matrix $\mathbf{X}^\intercal\mathbf{X}$. The projection matrix is computed using Cholesky decomposition. Whitening is equivalent to applying PCA followed by scaling [Murphy, 2012, p. 410].

6.3.2. Linear Discriminant Analysis

Linear Discriminant Analysis (LDA), or more precisely: Fisher's LDA – the most common type of LDA –, is a projection $\mathbf{z} = \mathbf{W}\mathbf{x}$ of a vector $\mathbf{x} \in \mathbb{R}^D$ to a lower-dimensional vector $\mathbf{z} \in \mathbb{R}^L$ using the $L \times D$ projection matrix $\mathbf{W}$. The transformation aims at maximizing the between-class variance while simultaneously minimizing the within-class variance[5]. Therefore, the cost function is defined as

$$J(\mathbf{W}) = \frac{\mathbf{W}^\intercal\mathbf{S_b}\mathbf{W}}{\mathbf{W}^\intercal\mathbf{S_w}\mathbf{W}} \tag{6.17}$$

where:
$\mathbf{S_b}$ is the between-class scatter matrix[6] defined as $\mathbf{S_b} = \sum_{k=1}^{K} N_k(m_k - m)(m_k - m)^\intercal$,
$\mathbf{S_w}$ is the within-class scatter matrix defined as $\mathbf{S_w} = \sum_{k=1}^{K}\sum_{n\in C_k}(x_n - m_k)(x_n - m_k)^\intercal$,
m_k is the class k empirical mean $\frac{1}{N_k}\sum_{n\in C_k} x_n$,
m is the overall empirical mean, and
C_k is the set of samples in class k.

The derivation of the above formula along with further explanations on how this can be reformulated as the eigenvalue problem $\mathbf{S_b} = \lambda\mathbf{S_w}$ can be found in [Bishop, 2006, p.188ff.] and [Murphy, 2012, p. 274ff.].

[5] LDA is closely related to PCA. Contrary to PCA, it searches vectors in the underlying space that best discriminate among classes rather than those that best describe the data (overall variance) [Martínez and Kak, 2001].
[6] The scatter matrix equals the covariance matrix if we remove the division factor [Bishop, 2006, p.189].

6.3.3. Within-Class Covariance Normalization

Within-Class Covariance Normalization (WCCN) [Hatch et al., 2006] is a projection $\mathbf{z} = \mathbf{W}\mathbf{x}$ such that $\frac{\mathbf{S_w}}{N} = \mathbf{W}\mathbf{W}^\intercal$ with $\mathbf{S_w}$[7] being the within-class scatter matrix as defined above for LDA. $\mathbf{W}$ is defined as the Cholesky factorization of $(\frac{1}{K}\mathbf{S_w})^{-1}$. As the name suggests, WCCN aims at normalizing the covariance within all classes.

6.4. Metrics

6.4.1. F1 Score

"The F1 score can be interpreted as a weighted average of the precision and recall, where an F1 score reaches its best value at 1 and worst score at 0. The relative contribution of precision and recall to the F1 score are equal."[8] The formula for the F1 score is:

F1 = 2 * (precision * recall) / (precision + recall) [Murphy, 2012, p. 185].

In the multi-class case, if not explicitly stated otherwise, the weighted F1 score is used. For that, we calculate metrics for each class and find their average, weighted by support (the number of true instances for each class).

6.4.2. Cosine Similarity and Correlation Coefficient

As the name suggest, the cosine similarity computes cosine of the angle between two vectors $\mathbf{x}$ and $\mathbf{x}'$ [van Dongen and Enright, 2012, p. 1]:

$$\frac{cov(\mathbf{x}, \mathbf{x}')}{\sigma_{\mathbf{x}}\sigma_{\mathbf{x}'}}. \tag{6.18}$$

If the vectors stand for realizations of discrete probability distributions and are centered, *Pearson's correlation coefficient* is obtained [van Dongen and Enright, 2012, p. 1, 2].

[7] `https://www.idiap.ch/software/bob/docs/releases/last/sphinx/html/trainer/generated/bob.trainer.WCCNTrainer.html#bob.trainer.WCCNTrainer`, accessed November 2nd, 2014

[8] Source: `http://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html`, accessed September 29th, 2015

6.4.3. Receiver Operating Characteristic and ROC AUC

The receiver operating characteristic curve (ROC) plots the true positive rate (sensitivity) over the false positive rate (1-specificity) for different decision thresholds. The area under the ROC, the ROC AUC, is a commonly used metric in machine learning for probabilistic binary classification [Fawcett, 2006] that characterizes the curve as a single number [Murphy, 2012, p. 183]. An ideal classifier's ROC curve would go from the lower-left to the top-left corner and then to the top-right corner, resulting in an AUC of 1.0 (100%), while a random classifier's curve would be a straight line from the lower-left to upper-right corner covering 50% of the available space [Richert, 2013]. ROC and ROC AUC measure the general performance of a classifier by covering a multitude of different decision thresholds instead of a single one.

6.4.4. Equal Error Rate and Half Total Error Rate

The equal error rate (EER for short) is the one point on the ROC curve at which the false acceptance rate and the false rejection rate are equal [Cheng and Wang, 2004, p. 1], i.e., $(P_{FA}, P_{FR}) = (EER, EER)$ or $FPR = FNR$ (*false positive rate* equals *false negative rate*). "Since $FNR = 1 - TPR$, we can compute the EER by drawing a line from the top left to the bottom right and seeing where it intersects the ROC curve." [Murphy, 2012, p. 183] "[EER] is reported in almost every publication on speaker recognition" [Brummer, 2010, p. 74] and has become the de-facto standard metric in the speaker recognition literature. It is attractive if in practice the cost of misclassification is equal for both directions.

Similarly, the Half Total Error Rate (HTER) is the average of the false positive and the false negative rate [Nautsch, 2014, p. 110].

7. Data Set Synthesis

The greater part of annotated corpora commonly used in any sub-discipline of speech processing are recorded under studio conditions and/or lack natural ambient sounds encountered in our daily lives. Self-recorded sets with suitable content, on the other hand, have to be annotated manually – a process that is infeasible for the many hours of speech needed in a representative audio corpus. To tackle this problem, we synthesized a proper data set according to our needs – an approach taken by others as well, e.g. [Eyben and Weninger, 2013]. In the following chapter we describe this process and its essential components. The aim for our dataset was to accurately simulate **conversations encountered in real-life scenarios, captured by a mobile phone**. Therefore, we combined annotated corpora (from speaker recognition and affect recognition) with real environmental background sounds ("noise") and, with the help of *impulse responses*, artificially modified the resulting audio signal to reflect imperfect recording conditions of a smartphone being carried around (cf. Figure 7.1).

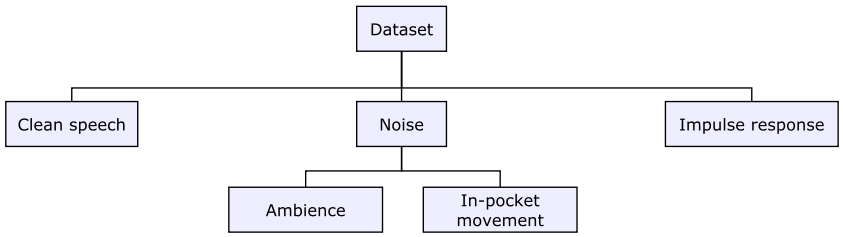

Figure 7.1.: *Overview: sources of audio for synthesizing the dataset*

7.1. Clean Speech

7.1.1. Speaker Recognition Data Sets

TIMIT The Texas Instruments/Massachusetts Institute of Technology corpus (TIMIT) [Garofolo et al., 1993] is a collection of read speech designed to provide speech data for acoustic-phonetic studies and for the development and evaluation of automatic speech recognition systems. "TIMIT contains broadband recordings of 630 speakers of eight

major dialects of American English, each reading ten phonetically rich sentences (16bit, 16kHz). 70% of the speakers are male, 30% female.

Buckeye The Buckeye Corpus of conversational speech [Pitt et al., 2007] contains high-quality recordings from 40 speakers in Columbus, OH conversing **freely** with an interviewer. The sessions were conducted as sociolinguistics interviews with everyday topics, and are essentially monologues. "The sample was stratified for age (under 30 and over 40) and sex, and the sampling frame was limited to middle-class Caucasians." [Pitt et al., 2005] The subjects spoke a total of 307,000 words adding up to about 26 hours of English speech. Audio files have a sampling rate of 16kHz and are stored in 16-bit.

VoxForge VoxForge[1] is an open-source project that provides a constantly growing, free, and annotated corpus of read speech in various languages including English and German. It is available in 44.1kHz, 16-bit format. While there is little meta information on speakers, a pitch-based clustering suggests a male-female ratio of about 3:1.

Mobio The MOBIO database [McCool et al., 2012] consists of bi-modal (audio and video) data taken from 152 people in five different countries. The free speech session from which we take our samples consists of answers to 10 random questions, all in the English language and recorded in non-studio settings (16kHz, 16bit). The database has a female-male ratio or nearly 1:2 (100 males and 52 females).

7.1.2. Emotion/Affect Recognition Data Sets

FAU Aibo The FAU Aibo corpus [Steidl, 2009] consists of 9 hours of German speech of 51 children at the age 10-13 years interacting with Sony's pet robot Aibo. This spontaneous, emotionally colored speech has been recorded via a close-talk microphone (16kHz, 16bit) and has been annotated with 11 emotion categories by five human labelers. We explicitly point out that this corpus is only used for selection and validation of suitable affect recognition techniques. As children's speech differs significantly from adult speech, the corpus does *not* serve for the creation of general affect models to be used with other data.

Semaine The Semaine corpus of emotionally colored conversations [McKeown et al., 2012] constitutes our main data set for affect recognition. It comprises recordings of 150 participants, for a total of 959 conversations between an agent and a participant – many of which were extensively annotated. In each recording the voice of the agent features a certain emotion with the goal of eliciting a particular *natural* affective reaction from the participant. 6-8 raters per clip then traced five affective dimensions and 27 associated categories.

[1] `http://www.voxforge.org/`, accessed November 15, 2014

7.2. Noise

Given our aim to simulate real-life situations as captured by a potentially ubiquitous smartphone, our dataset must not only contain speech but must also reflect the acoustic characteristics of the user's assumed environment. To be able to further specify these characteristics, we assume the exemplary target user to live and work in a major western or central European city. Our selection of sound categories for this environment, also referred to as sound spaces, is founded in the literature, logical deduction and empirical observation.

In [Yang and Kang, 2005, p. 214ff.], the authors identified the following main sound sources in 14 **urban open public spaces** across cities in Italy, Greece, Germany, UK, and Switzerland: **traffic, surrounding speech, children, construction, water, demolition, church bells, footsteps, and birds**. "In terms of function, the sites included residential squares (e.g. Kritis Square, Petazzi Square, Jardin de Perolles), cultural and tourism squares (e.g. the Seashore of Alimos, Peace Gardens, Barkers Pool, All Saint's Garden, Silver Street Bridge, Florentiner Square), railway station squares (e.g. Bahnhofsplatz, Place de la Gare), and multi-functional squares (e.g. Karaiskaki Square, Makedonomahon Square, IV Novembre Square)." [ibd.]

In contrast to these urban places, [Szeremeta and Zannin, 2009] had 335 participants characterize the *soundscape* (etym.: sound + landscape) of more natural public areas in terms of four public parks. Even in such a setting the sound of traffic, besides speech, is predominant: "The sounds of **birds, vehicle traffic, other natural sounds** and people were identified regularly in the area of the 4 parks, making up a total of 89.9% of the sample and confirming that these are the principal sounds that make up the soundscape of these areas. Thus, 32.6% of the references involved birdsong, 28.5% vehicle traffic, 15.8% other natural sounds, and 13% referred to human sounds [...]." [Szeremeta and Zannin, 2009, p. 6146]

Other authors classify everyday sounds according to activities they relate to. [Díaz and Pedrero, 2006] tried to determine average sound exposure by activity. For that they conducted a cross-section study in Madrid, Spain where 35 participants had to constantly wear a so-called *noise dose meter* for one week. A broad range of activities was observed yielding the following sound environment categories: **leisure** (sports, social gatherings etc.), **education** (classroom, meeting, library etc.), **domestic** (music, TV etc.), **occupational** (warehouse, office, workshop etc.), **shopping**, sleep, and **transportation** (c.f. Figure 7.2). Similar sound spaces but different contribution rates were identified by [Mehl and Pennebaker, 2003] for the population of undergrad college students, underlining the validity of the model.

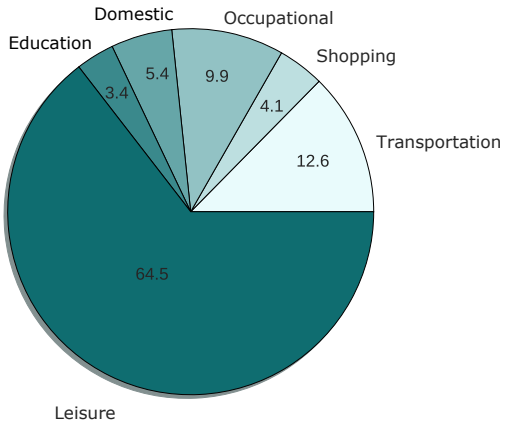

Figure 7.2.: *Daily sound exposure according to activity [Díaz and Pedrero, 2006, p. 278]*

Based on the above literature we defined the following categories of ambient sounds which we jointly refer to as *noise*:

- Outdoor, street, nature, traffic *,**
- Bar, cafe, crowd *,**
- Public transportation *
- Indoor (living room, office, kitchen, bathroom etc.)
- In-car
- Shopping *
- Pocket movement
- Foot steps

Categories marked with a single star had to contain babble sounds (incomprehensible speech). Two stars mark the occasional presence of music. While not mentioned in the literature, we added one sound category that we deem essential in our scenario: sounds of the smartphone moving in the pocket, creating friction with surrounding textile.

Samples for the above sound spaces were then obtained from the following sources:

QUT-NOISE The corpus is "designed to enable a thorough evaluation of voice activity detection (VAD) algorithms across a wide variety of common background noise scenarios" [Dean and Sridharan, 2010, p. 1]. From it we selected sound files that matched the above categories and that did not contain intelligible speech.

freesound.org Freesound is a collaborative database of Creative Commons Licensed sounds that provided samples of foot steps and various public scenarios for our experiments.

Own recordings For a Master's thesis project [Seitle, 2014], we asked two students

from Munich to record and label their respective environment on two typical workdays (full days including after-work activities). They were instructed to discard any segments that contained intelligible speech. Recording was conducted using a Samsung Galaxy S3 mini smartphone and an Apple iPad 2.

All audio files were grouped according to sound category. The combined noise set covers all general sound spaces found in relevant literature and comprises 96 minutes of audio in 16kHz, 16-bit format.

7.3. Impulse Responses for Channel Effect Simulation

7.3.1. Motivation and Data

Much of the more recent literature in speaker recognition concerns itself with the compensation of so-called channel effects, i.e. signal alterations caused by varying microphones for training and testing as well as other factors "such as [...] environment [...] and transmission means (e.g., landline, cellular, VoIP, etc.)." [Reynolds, 2003, p. 1] In the context of ubiquitous smartphones this also encompasses signal alterations induced by the characteristics of the position of the recording phone. Note that background sounds/noises are signals themselves and do not constitute channel effects.

To further specify the kinds of channel effects relevant for our dataset, we again consulted the literature: [Ichikawa et al., 2005] and [Reimers, 2010] both conducted gender-balanced surveys in major Western cities (Helsinki, Milan, New York; Munich) to find out where "mobile phones are carried whilst users are out and about in public spaces" [Ichikawa et al., 2005, p. 1]. **Pockets and bags/backpacks** account for 90% of the answers in each survey.

In signal processing, channel effects can be represented as so-called *impulse responses* (IR). An IR describes how an abstract system alters an input signal [Pesaran and Shin, 1998, p. 18]. In our case, the entirety of textiles around the smartphone and also the smartphone frame form such a system. We can calculate its corresponding impulse response experimentally by comparing a known input signal with its system-recorded output. As every fabric covering the microphone evidently alters the captured audio signal in a different, characteristic way, we picked one representative for each of the two above categories: denim jeans and a leather bag. Both exemplars are assumed to be common in the target population and also to be thick enough to presumably cause high attenuation and alteration of the signal. As we hypothesized that the orientation of the smartphone would have a measurable effect on the impulse response, we recorded two orientation-specific IRs within a denim jeans pocket: one with the

device in an upwards orientation (microphone towards the top), one in a downwards orientation (microphone towards the ground). Since the studies mentioned above focused on public spaces, we included a presumed category for private spaces: the smartphone lying on a desk. In summary, four representative IR categories were established.

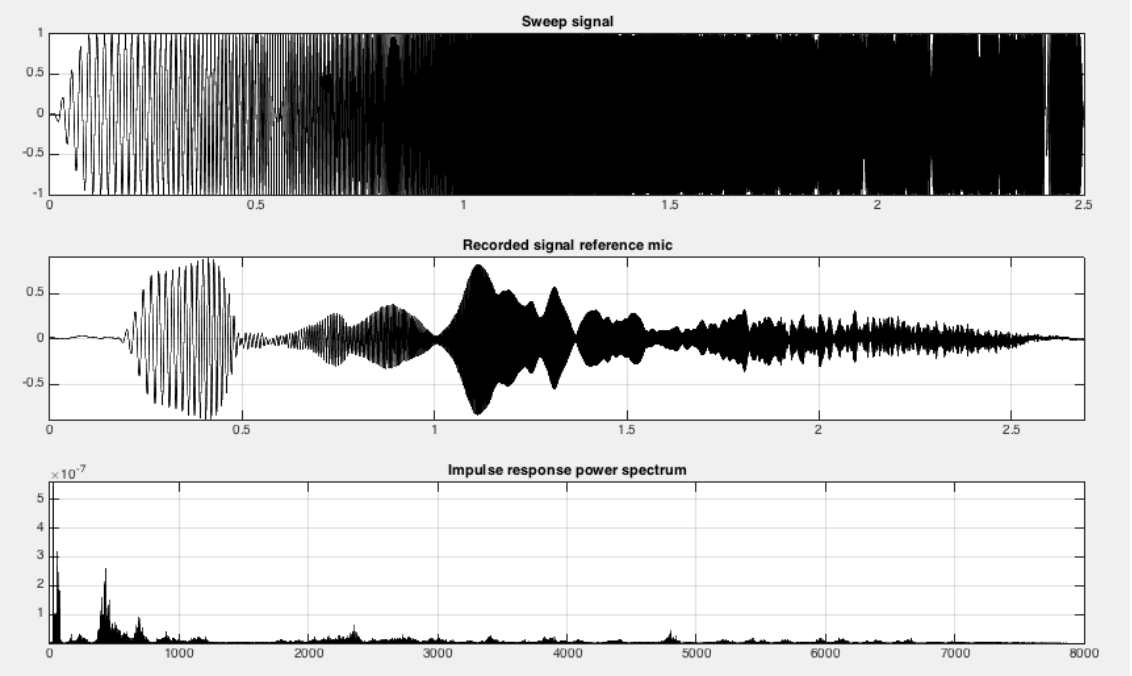

Figure 7.3.: *Sweep signal (40Hz-20kHz over 2.5s), signal recorded by a reference microphone, and spectral density of the resulting impulse response.*

All audio recording was carried out at TU Munich inside a sound-proof, decoupled, low-reflection recording booth with calibrated equipment. On a Dynaudio BM6A MKII near-field monitor speaker we played a so-called *sine sweep*, a tone with rising frequency, ranging from 40 to 20,000 Hz (cf. Figure 7.3, top row) which we recorded on the smartphones in four positions. By deconvolving the source signal and the recorded signal we obtained the target impulse response.

7.3.2. Power spectra and similarity of impulse responses

In Figure 7.4 an exemplary spectogram (energy in the frequency spectrum over time) of a sine sweep recorded with the LG G2 smartphone in four positions is plotted. Over the duration of the sine sweep (horizontal axis) the frequency (vertical axis) rises from 0Hz on the far left to

8kHz[2] on the far right. White parts mark very high energy, red marks medium to high energy, and blue denotes low energy. Attenuation of energy caused by fibre is clearly observable as white parts turn red in the according spectrograms, especially in high-frequency space. As would be expected, there is little attenuation when the smartphone is placed on the table.

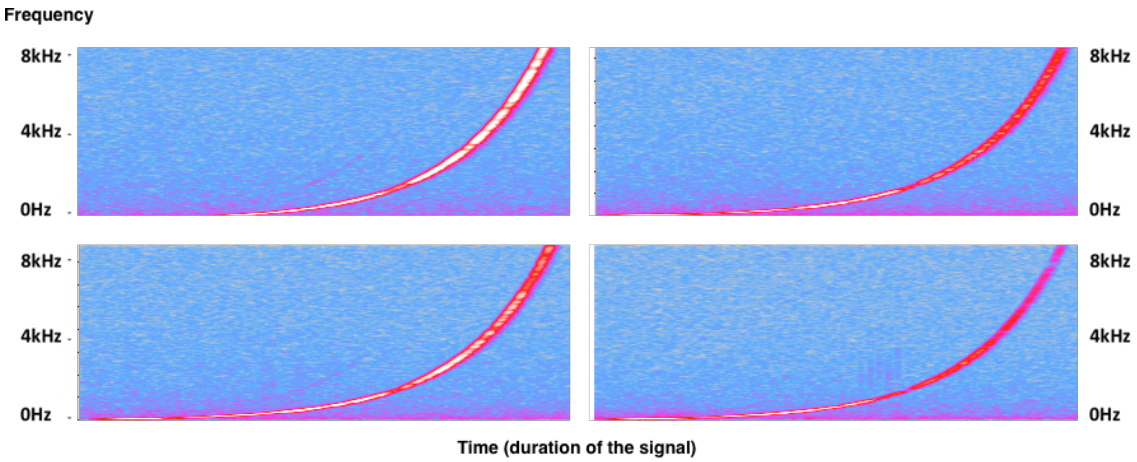

Figure 7.4.: *Spectrogram of the recorded sine sweep for the LG G2 in the positions: table (upper left), pocket with microphone upwards (lower left), pocket with microphone downwards (upper right), leather bag (lower right)*

Figure 7.5 depicts the *power spectral density* [Moses and Randolph, 2004, p. 6] (energy per frequency, integrated over time) of all 16 impulse responses (4 smartphones, 4 positions, respectively). The differences in distribution of energy over the frequency spectrum, i.e. the *channel effects*, are evident both between smartphones and between positions. Inter-position variance is more dominant than inter-smartphone variance. We see that for the most part only low frequencies are able to penetrate objects (e.g. fibre) between the source of the signal and the recording device. For devices carried, for example, in a leather bag, most of the energy of the original audio signal is attenuated.

[2]The sampling rate of the recording microphone was 16kHz, thus the maximum frequency in the signal is 8kHz.

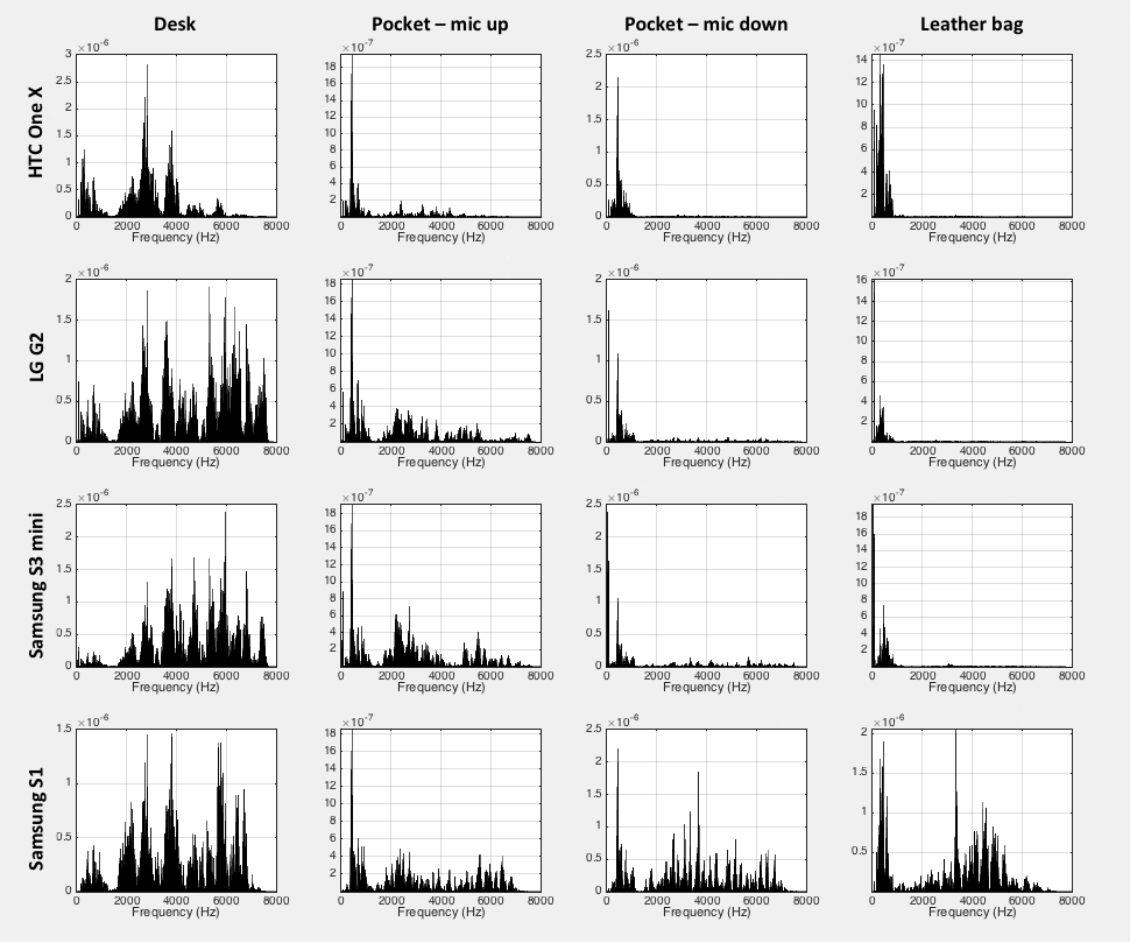

Figure 7.5.: *Periodogram (power spectral density) of recorded impulse responses for four smartphones in four positions, respectively.*

To further compare the above channel effects *in the domain of speech processing*, we analyzed the effect of our impulse responses on our main set of features in all speech processing tasks: MFCCs. 10 seconds of white noise served as a reference signal from which MFCC vectors (18 coefficients + 1st-order time deltas, 20ms frames, 10ms overlap) were computed. In a white noise signal, all noise is entirely random and hence, roughly, of equal energy over all frequency ranges. The same white noise input signal was convolved with each of the various impulse responses, the output then compared using cosine similarity. An alternative metric would be correlation which is also based on the inner product of the vectors and thus equivalent for centered data. In subsequent chapters we will quantify the impact of channel effects on *classification performance*. At this point, we are interested in relative similarities. Thus, for readability, we standardized all similarity scores with their mean and standard deviation (computed over all $\binom{16}{2} = 120$ pairs).

Standardized intra-class similarity scores for smartphones are listed in Table 7.1. For each smartphone, all position-specific feature vectors were compared with one another. For each pair of positions, the absolute value[3] of the cosine similarity was averaged over time, i.e. all frames. Eventually, the mean similarity over all position pairs was computed as the phone's intra-class similarity score. It quantifies how similar the position-specific channel effects of the target smartphone are compared to the overall similarity of all measured channel effects. Analogously, Table 7.2 shows intra-class similarities for phone positions denoting how similar different phones are when carried in the target position.

Our findings from the periodigrams above are reflected in the MFCC features. From the two tables, one can draw the following conclusions for speech processing:

- On average, the channel effects caused by **different positions are more dominant** than the ones caused by different phones (relative signal similarity 0.21 and 0.41, respectively).
- There is an obvious order of effect magnitude: the more a phone is surrounded by fibre, the less similarity of recorded speech signals there is. The majority of impulse responses represent positions in which the phone is covered. On a desk, i.e. without obstacles in the way of the speech wave, signals are much more similar than on average (1.2 standard deviations above the mean).

HTC One X	−0.1149
LG G2	−0.2364
Samsung S1	1.2304
Samsung S3 mini	−0.0203
Mean	0.2147

Table 7.1.: *Standardized intra-smartphone (differences between positions) cosine similarities between MFCC vectors with channel effects.*

Desk	1.2270
Pocket (microphone upwards)	0.6562
Pocket (microphone downwards)	0.0978
Leather bag	−0.3233
Mean	0.4144

Table 7.2.: *Standardized intra-position (differences between smartphones) cosine similarities between MFCC vectors with channel effects.*

[3]Note that the cosine similarity is symmetric and in the range $[-1; 1]$ with 0 marking orthogonality (no correlation).

The Samsung Galaxy S1 smartphone exhibits an exceptionally high intra-class similarity. In the above computations, we assumed this somewhat strange phenomenon to accurately reflect reality. It suggests that the microphone of this low-end phone and/or its frame are the deciding factors, "averaging" any input signal compared to other phones with supposedly higher quality components/manufacturing.

7.3.3. Approximation of impulse responses

The above impulse responses were extracted in a professional manner – a time- and resource-intensive process that is inadequate for ad-hoc experimentation. While not the focus of this thesis, we want to give short consideration to the idea of approximating impulse responses. If successful, this would allow for much easier incorporation of emulated channel effects in future research. The findings of this section also bear great relevance for smartphone-based audio processing in general.

For the approximation, we replicated the sine sweep procedure described above in a $15m^2$ office room with standard computer equipment, i.e. multimedia sound speakers, and the same smartphones as before. Positions used in the professional extraction were reconstructed as concordantly as possible. Results are depicted in Figure 7.6.

It is evident that resemblance to the impulse responses in Figure 7.5 is superficial at best. We validated this observation by computing cosine similarities of the resulting MFCC as done in Section 7.3.2. Results are not listed here to avoid redundancy. We identified three differing factors and thus three potential reasons for this dissimilarity: inadequate *speakers*, an inadequate room in terms of *reverberation*, and the possibility that there are no archetypal *positions* that allow for positional profiles, i.e. minimal changes in position or environment (clothing fabric) have great effect on the signal.

With further experimentation we were able to rule out the first two variables as major contributors.

First, we compared the low-end speakers to acoustically decoupled high-end multimedia speakers (Mackie CR3). The same sine sweep was played consecutively on both speaker and recorded on an LG G2 smartphone in 30cm distance. Neither the smartphone nor any other object was moved during the experiment. Impulse responses of the two sweeps are shown in Figure 7.7. Energy distribution over the spectrum is very similar (cosine similarity 1.31 standard deviations above the mean similarity, cf. Section 7.3.2).

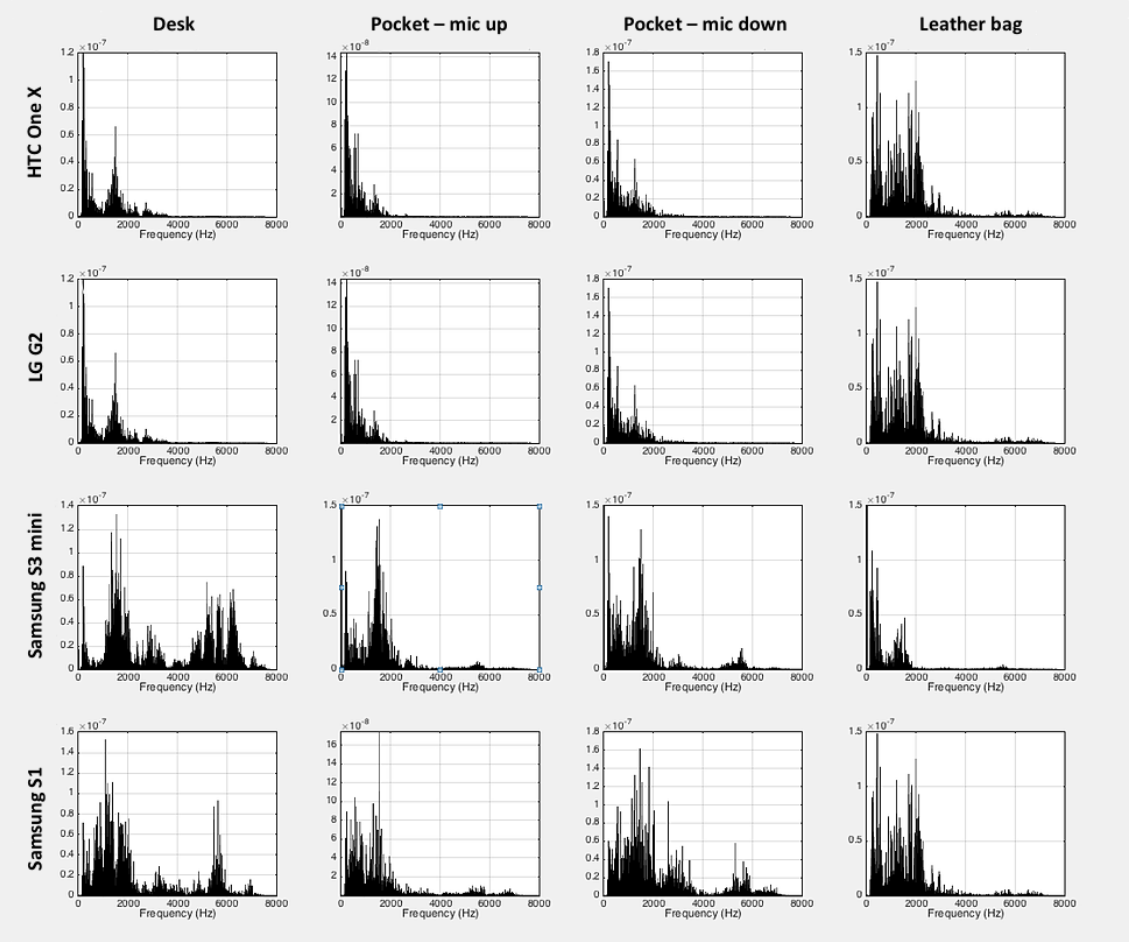

Figure 7.6.: *Periodogram (power spectral density) of approximated impulse responses. Little resemblance to real IRs (cf. Figure 7.5).*

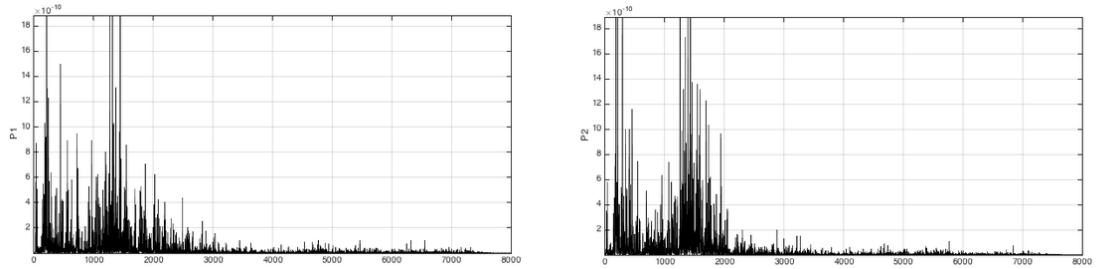

Figure 7.7.: *Power spectral density comparison for low-end and high-end speakers as sound source. Very little difference can be observed.*

To identify reverberation in the room we again look at spectrogram of the original impulse response and compare it with the spectrogram of the approximated IR. While there is some reverberation present in the signal (“smear effect”), the effect is small and pertains only to fre-

quencies higher than 2kHz. Most of the mismatch between the two sets of impulse responses can be found below that mark where the majority of energy is located (see Fig. 7.7). Note that the color disparity is caused by different speaker volumes for the two recording sessions.

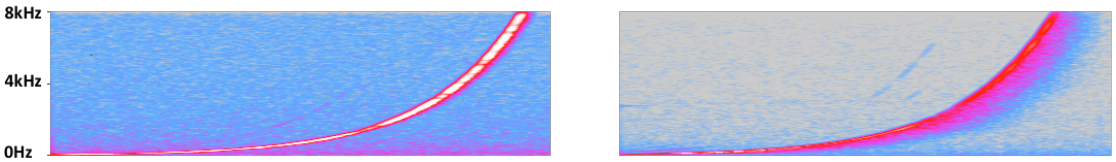

Figure 7.8.: *Comparison of spectrograms for real and approximated impulse response (exemplary). Reverberation only noticable above 2kHz.*

We conclude that most of the variance can be attributed to imperfect recreation of the original position. **This underlines our findings from Section 7.3.2 and suggests that it is pointless to create position profiles because small variance in position has too great an effect on the signal to construct universal positional fingerprints.** For our use-case, this also invalidates Dunn/Reynold's proposition [Dunn et al., 2001, p. 1ff.] of using handset-specific profiles in speech processing as effects induced by positional variance, when present, occlude smartphone-specific characteristics.

7.4. Audio Signal Mixing

After our thorough examination of channel effects, we get back to the main matter of this chapter: the synthesis of a suitable dataset. In the preceding sections we have described the individual components: speech, ambient noise, and impulse responses for the simulation of channel effects. The according pools of audio files were combined in a procedure we denominate signal *mixing* (cf. Figure 7.9). Every audio file (snippet) from the clean speech corpora was processed exactly once. We iterated over all clean speech snippets until each one had been processed. As the final dataset should contain some segments with and some without noise, all mixing was probabilistic. With a certain probability the clean speech snippet in question was mixed (superimposed) with other speech snippets and/or noise randomly drawn from the two noise pools (ambient noise and pocket movement). Before being mixed together, all snippets were convolved with the same randomly drawn impulse response to simulate recording conditions encountered in real-life.

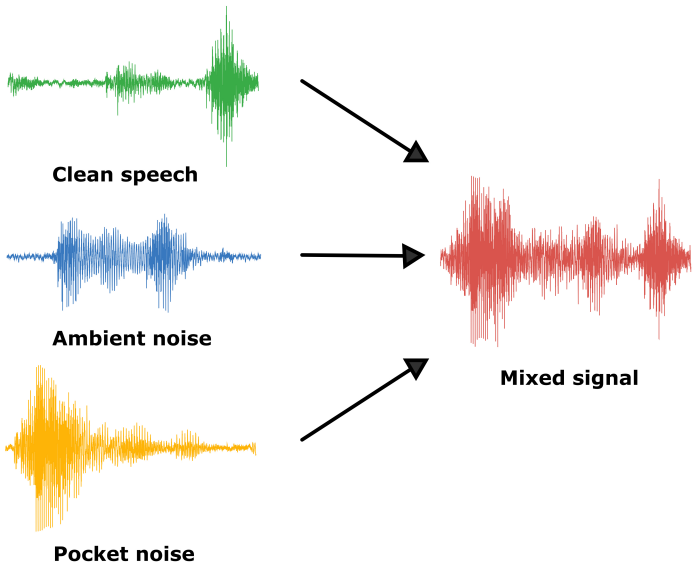

Figure 7.9.: *Signal mixing: superimposing audio segments to produce a noisy speech corpus*

In greater detail, the procedure consisted of the following steps which were repeated until every audio file of the speech corpus/corpora had been processed. The parameters were chosen in accordance with [Eyben and Weninger, 2013, p. 3].

Produce clean speech segment containing multiple speech snippets

A random number $n \in [1;5]$ of audio files was drawn without replacement from either the entire pool of clean speech files or a speaker-specific pool – depending on the experiment. Each snippet was re-sampled to the desired sampling rate *fs* and convolved with one of the impulse responses. The latter was either selected a priori or picked at random depending on the experiment. The amplitude of each speech signal was normalized to the same loudness level according to the Replay Gain standard [Robinson, 2001], [Nygren, 2009] and then multiplied with a segment-wide random gain factor $g_{seg} \in [+3\text{dB}; -20\text{dB}]$ to simulate different real-life situations with respect to volume. This also reflects different distances to the recording smartphone. As hard cuts would generally result in artifacts in the Fourier spectrum [Harris, 1978, p. 1ff.], linear fading over 1000 samples (in and out) was applied to the speech snippets. To ensure the feasibility of fading, audio files shorter than 2000 samples were generally omitted. Afterwards, the individual snippets were concatenated sequentially to a single combined speech signal.

One of the real-life scenarios to be simulated with the dataset was two or more people talking at the same time. For instance, if one speaker would try to interrupt the other

or both would start talking at the same time. Another likely real-life scenario is the occurrence of an unrelated conversation in the background in which the owner of the smartphone is not engaged. To account for such scenarios we defined a maximum of three simultaneous speakers – not counting unintelligible babble of voices – at any given time. In addition to the aforementioned segment-wide gain, a random speaker-specific attenuation $g_{overlap} \in [0\text{dB}; -10\text{dB}]$ was applied to overlapping speech snippets so that the speakers varied in their loudness. The probability of an overlap for two consecutive snippets $p_{overlap}$ was set to 0.44, the empirical percentage of overlaps at turn transitions in face-to-face conversations in [Ten Bosch et al., 2005, p. 84], with the starting point of the new, second signal randomly placed within the first, i.e. the concatenation of all preceding, possibly overlapping snippets in this current segment. If for two consecutive snippets there was no overlap, a random amount $s \in [0.5\text{s}; 5\text{s}]$ of silence was embedded in between. For that, we recorded 3 minutes of silence – as opposed to using a zero-amplitude signal – and found it to be equivalent to pink noise. Whenever the limit for concurrent speech signals was reached, a subsequent period of silence was enforced. In that case, the earliest potential starting point for any follow-up snippet was the end of the silence period. At first we chose the length of silence based on empirical data of real-life conversation. During the course of our experiments, however, we decided not to consider temporal dependencies (cf. 9.1). Therefore, the final parameters as they are listed here are based on the *desired* overall ratio of silence to speech in the dataset which is roughly 60/40. This value should reflect reality and is based on reports by [Mehl et al., 2001, p. 522] who conducted a study for which participants were equipped with a recording device that captured acoustic samples over the course of four days in the participant's everyday life. The reported fractions of speech presence ($0.26+0.34 = 0.6$) in [Mehl et al., 2001] refer to waking time (16h) and were weighted accordingly ($\frac{16}{24}$) under the assumptions that people sleep in silent environments.

The annotation of the individual snippets was merged accordingly.

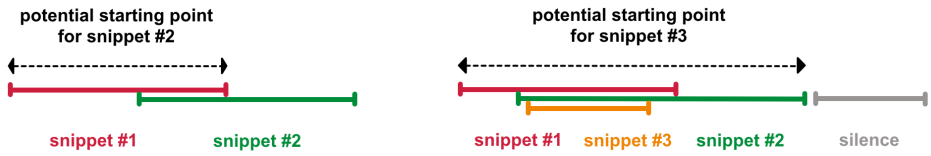

Figure 7.10.: *Two (unrelated) scenarios illustrating the mixing procedure: two speech snippets are joined together with overlap (left); three snippets overlap one another followed by an enforced period of silence (right).*

Produce noise segments and mix with speech segment

With the probability $p_{ambient} = 0.56$ the entire speech segment of the previous step was mixed with ambient noise. If so, a segment of the same length as the speech segment was drawn from the pool of ambient noises. As the various short noise files had been concatenated to longer noise class specific files, a segment of the desired length was easily obtainable. We base $p_{ambient}$ on an upper bound in terms of the fraction of daily activities that (arguably) require the absence of background noises (reading, sleeping, etc.) as listed in [Mehl and Pennebaker, 2003, p. 861] and [Mehl et al., 2001, p. 522][4].

Irrespective of any ambient noise, an additional probability $p_{pocket} = 0.055$ determined whether the characteristic sound of movement of the smartphone in a pocket would be present in the final segment. It was drawn with replacement as well. Our choice for p_{pocket} is founded on an average daily step count of 6,500 for US adults reported in [Tudor-Locke et al., 2010]. With a walking speed of 100 steps per minute (moderate-intensity walking [Marshall et al., 2009]) this translates to 65 minutes per day (5%). Similar numbers are given by [De Cooman, 2015]. Across Europe, this number is consistently 10% higher than in the US according to the latter source, leading to a probability of 0.055.

Every noise signal was convolved with the impulse response, its amplitude first normalized to equal loudness, then multiplied with the same gain factor as the speech segment. At that point the speech segment and its noise segment counterpart had the same loudness. Eventually, linear fading was performed before speech and noise were superimposed with a random signal to noise ratio $SNR \in [-2\text{dB}; 20\text{dB}]$.

Due to the summing up and scaling of amplitudes, it was possible for any final mixed segment to contain samples with an amplitude that exceeded the limits imposed by the data type. In practice, however, this rarely happened. To still avoid hard clipping of an affected sample, we post-processed all mixed audio files with dynamic range compression and reduced high peaks with very short attack and release times (−10dB threshold, 1 : 5 compression ratio, 5ms attack time, 10ms release time).

The above procedure was slightly modified for the data subset used to train and evaluate individual speaker models. Whenever speech segments did not allow for the presence of multiple speakers, the pool of speech files from which the snippets were drawn was confined to the target speaker's file pool.

[4] 8h of sleep translate to 33%; 16.4% for working and reading refer to 16h waking time and make up for $0.164 \cdot \frac{16}{24} = 11\%$ overall

7.5. Limitations

While we are confident that our dataset is a realistic representation of every-day environments and scenarios, we are aware of three limitations that need to be mentioned:

- Spatial characteristics like reverberation in closed environments have not been considered. Recent literature reports that state-of-the-art speech processing algorithms are able to cope with reverberation [Sadjadi et al., 2012], [Boil and Hansen, 2010].
- As is the case with any synthetic dataset, the so-called Lombard effect ("speakers alter their style of speech in noisier conditions in an attempt to improve intelligibility [Woo et al., 2006, p. 1]) is not reflected in our data.
- The drawing of noise segments from the various noise categories was entirely random (equal probability) and didn't account for the different percentages/likelihoods with which the respective category is encountered in our daily lives. This limitation is due to a lack of consistent data from the literature w.r.t. percentages and also due to the fact that, intuitively, in some environments conversations have a higher probability of emerging than in others. Our dataset focuses on conversations and disregards other parts of people's lives. In our case, obtaining perfectly representative data without an extensive dedicated study is nearly impossible. Thus, we considered it most important that any noise category be present in the data, regardless of its share.

8. Audio Signal Processing

In the previous chapter we have presented our methodology for obtaining a dataset suitable for our needs. Before we evaluate current state-of-the-art algorithms for recognizing speech, speakers, and affect, we provide the reader with an overview of how constant audio streams are (pre-)processed and discretized for further analysis.

First, acoustic sound must be captured digitally. Almost every current-generation mobile operating system provides means in form of an API to obtain a live audio stream of the target sampling rate using the device's built-in microphone. For that, the OS keeps filling a buffer from which the stream can be read in "chunks" of waveform samples. Our subsequent processing of the signal follows the standard approach unanimously described in the literature (e.g. [Kinnunen and Li, 2010], [Togneri and Pullella, 2011], [Beigi, 2011]). The procedure is illustrated in Figure 8.1.

Pre-emphasis A high-pass filter is applied to the array (buffer) of waveform audio samples. "This emphasises the higher frequencies and compensates for the human speech production process which tends to attenuate high frequencies." Thus, we use a 1st-order high-pass filter with the common coefficient value 0.97.

Framing The time-domain signal is then divided into segments of uniform length called *frames*. Subsequent frames may overlap if their offset, the so-called *frame increment*, is shorter than their length/size. Typically, frame lengths range from 20ms to 32ms with offsets starting from 10ms to frame length, i.e. without any overlap. [Togneri and Pullella, 2011, p. 27], [Wöllmer, 2013, p. 21]

Windowing To avoid artificial aliasing effects in the frequency spectrum introduced by cutting the signal into segments, we smoothen the edges of frames by multiplying them with a windowing function like, e.g., a Hanning or a Hamming window [Blackman and Tukey, 1959]. It must be noted that the term *window* is used ambiguously in speech processing as it also refers to a set of subsequent frames. If no clarification is provided, usually the latter meaning is intended.

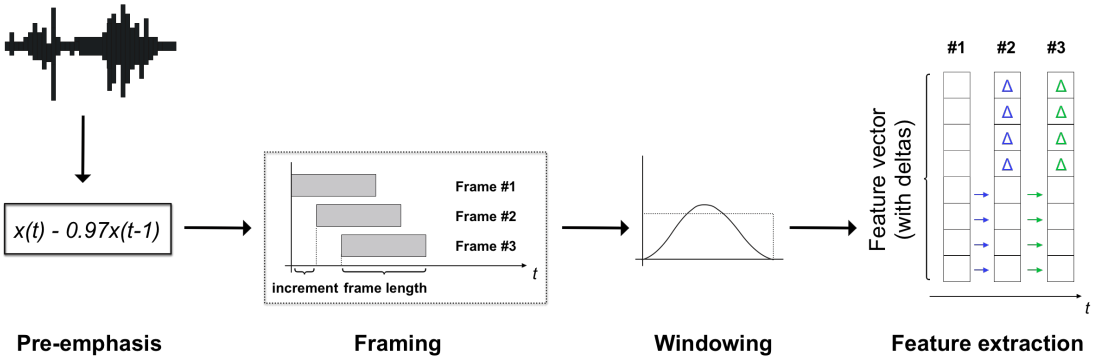

Figure 8.1.: *Signal processing steps: The sampled audio signal gets high-pass filtered and segmented into potentially overlapping frames – depending on frame size and increment. A smoothing window function is applied to prevent spectral aliasing. Afterwards, features are extracted and concatenated on a frame-by-frame basis. Along with concatenated temporal derivatives they form feature vectors. Illustration based on [Togneri and Pullella, 2011, p. 27] and [Kinnunen and Li, 2010, p. 15].*

This is followed by domain-specific *feature extraction*. Once the frames are represented by feature vectors, they are passed through the speech processing pipeline to establish a live conversation model. This pipeline consists of three cascaded steps that we will cover in the subsequent chapters.

1. Detect speech in a given audio stream using a suitable method from the domain of *Voice Activity Detection* (Chapter 9).
2. Detect whether the speech/voice belongs to the target speaker, i.e., whether she is engaged in a conversation (*Speaker Recognition*, Chapter 10).
3. Analyze the character of the conversation using methods from *Emotion/Affect Recognition* (Chapter 11).

9. Voice Activity Detection

In this chapter, we cover the algorithmic detection of speech in a given audio signal. We first present an overview of established and recently reported methods. Note that these methods haven't been tested on audio signals that contain channel effects. Therefore, we conduct our evaluation in two steps: as we tackle this task like a general classification problem, we first assess performance by including the same channel effects in training and testing. This will give us a notion of how well the original task is performed on a realistic signal. As a second step, the impact of *unknown* channel effects is studied.

We will conclude that most sophisticated general-purpose classifiers perform similarly well with both known and unknown channel effects present (ROC AUC > 0.9) and that MFCCs alone constitute a sufficient feature.

9.1. State-of-the-Art Methods

Voice activity detection (VAD), i.e., the discrimination of speech in a given signal, is an important step in almost every speech processing pipeline [Sakhnov et al., 2009, p. 1], especially in speaker recognition for filtering out segments that do not contain speech. While energy based systems provide high accuracy in the absence of noise, performance degrades substantially in low-SNR (signal-to-noise ratio) environments [Ramirez et al., 2007, p. 6]. Recent literature commonly lists a handful of approaches for VAD in the presence of non-stationary noise. Most frequently, [Ramirez et al., 2004] and [Sohn, 1999] are cited. The former calculate the so-called Long-Term Spectral Divergence (LTSD) between speech and noise and compare it to a dynamically adapted decision threshold. The latter propose a statistical approach based on the likelihood ratio (LR) test and Hidden Markov models (HMM). Several authors, e.g. [Kinnunen et al., 2007] and [Dean and Sridharan, 2010], regard VAD as an ordinary classification problem to be tackled by all-purpose classifiers like GMMs and SVMs. Most recently, [Eyben and Weninger, 2013] employed Long Short-Term Memory Recurrent Neural Networks (LSTM RNN) to shift focus on temporal dependencies. While there is no comprehensive evaluation of all approaches in direct comparison, [Kinnunen et al., 2007], [Dean and Sridharan, 2010] and [Eyben and Weninger, 2013] show that LTSD- and LR-based approaches are clearly out-

performed by the more recent algorithms, particularly in terms of consistency.

Apart from spectral divergence, VAD *features* commonly found in the literature include spectral entropy, zero crossing rate, cepstral features (MFCCs, PLPs) and pitch (cf. Section 6.1).

Table 9.1 gives a summarizing overview of state-of-the-art VAD methods and their relevant properties for this work. For more specific information on performance we refer the reader to the above evaluations.

Approach	**Time dependency**	**Performance and robustness**
Adaptive LTSD	yes	varying
LR HMM	yes	varying
LSTM RNN	optional	good (0.1 EER/0.96 AUC at [-6;25]dB SNR)
All-purpose classifiers	no	good (0.15 HTER at 0dB SNR)

Table 9.1.: *Common VAD approaches for noisy environments. Performance reports pertain to different conditions (signal-to-noise ratios).*

Time dependent approaches require a training set that correctly reflects the temporal aspects of real conversations. Although we tried to mimic these aspects in our dataset synthesis, it may contain artificial dependencies that could falsify the results. Therefore, we consider only universal time independent classifiers and restrict the LSTM to a single timestep, removing its dependency. We will show (cf. Section 9.3.2) that this decision does not significantly impact performance as our results are on par with those of time dependent classifiers as reported in the literature.

9.2. Performance evaluation

9.2.1. Methodology

To evaluate voice activity detection on smartphone-recorded audio we allocated VAD subsets from the corpora TIMIT and Buckeye, pooled the subsets and mixed them with noise as described in Chapter 7. Each audio snippet was convolved with a randomly drawn impulse response. 200,000 frames were drawn randomly for this experiment. 38.34% represented speech. Although our corpus would have allowed for a bigger dataset, the potentially cubic complexity of support vector machines imposed a practical constraint on our efforts to maximize validity. Each frame was represented by a 36-dimensional feature vector of 18 Mel and 18 delta Mel coefficients. All dimensions were normalized to standard score using the first two moments (mean, standard deviation) over the entire training data (dimensions are already

decorrelated, cf. Section 6.1.4). Training and testing was performed frame-wise. Classifiers considered for VAD experiments were: SVM (full and stochastic gradient descent), Random Forest, LSTM, GMM, Naive Bayes, Logistic Regression (cf. Section 6.2).

To determine the number of components (Gaussians) for the Gaussian mixture model we drew on our previous results from [Seitle, 2014] where we found 8 components to be suitable. We then examined according density plots of all features for both classes (speech and non-speech/other). As the exemplary plots of the first six MFCCs depicted in Figure 9.1 indicate, the *individual* distributions can be approximated well by this small number of superimposed Gaussians. In [Seitle, 2014, p. 99] we demonstrated that increasing the model's complexity resulted in comparatively small performance gains and from 16 components upwards in considerable variance. To avoid the risk of overfitting and to minimize computational complexity we therefore decided to go with the 8 components per model. Class priors for the GMM were estimated from the dataset.

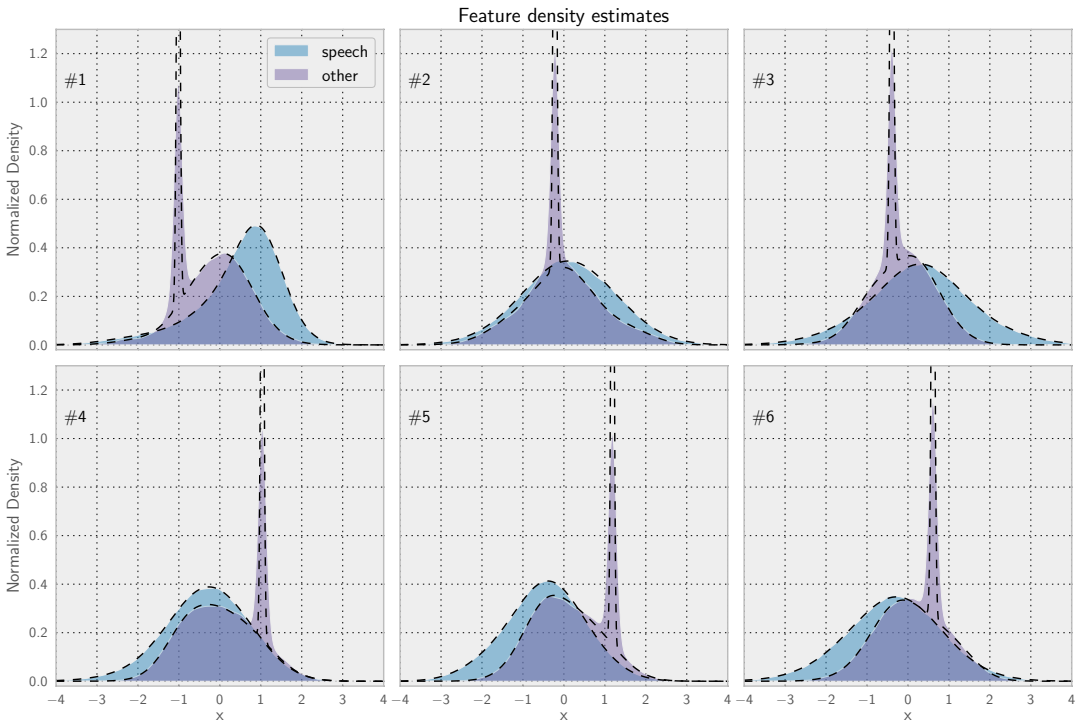

Figure 9.1.: *Probability density functions of the first six Mel Frequency Cepstral Coefficients (Gaussian kernel density estimation) of the training data. Dashed lines are approximations by individual 8-component GMMs that evidently fit the target distribution adequately.*

Another observation can be made in Figure 9.1. While the distribution of non-speech MFCCs bears remarkable similarity to the speech-specific distribution, there is a distinctive spike in every plot. The most likely explanation for this phenomenon lies in the dichotomy of non-speech frames: while simple silence (partly pink noise) shows little spectral variance (spikes), ambient background noises can fall into any spectral band.

Past experiences with random forests in voice activity detection [Seitle, 2014, p. 87] indicated that the performance gain of random forests for more than 10 trees is negligible compared to computational demands. Consequently, this setup was kept for our experiments.

As [Eyben et al., 2012, p. 17] briefly discuss, there is no established topology for LSTMs in speech processing. We therefore follow their approach of restricting ourselves to a small set of different topologies among which we choose for parameter selection. Single-layer configurations in our set comprised 10, 25 ($\frac{2}{3}\ of inputs + outputs$, [Sarle, 2002]), 50, 100, and 144 ($4 \cdot$ inputs) hidden units. Two-layer configurations were of the following sizes: (10, 20), (20,10), (100, 10), (10, 100).

For each classifier 10-fold cross-validation (CV) was conducted to compute statistically sound performance estimates with respect to generalization. The average performance over all folds was reported for comparison. If model parameters had to be determined, additional 10-fold CVs were executed for every possible parameter combination by means of nested cross-validation [Varma and Simon, 2006]: in each of the ten outer folds the training set served as basis for selecting the optimal parameter configuration. If m possible configurations had to be evaluated, m inner 10-fold CVs were run on the training set of the outer fold. The regular performance estimation of the outer fold was then conducted using the best of the m configurations. [Salzberg, 1997, p. 325]

Most classifiers described in Section 6.2 are probabilistic and thus return a real-valued probability as opposed to just a class label. Thus, for a binary classification problem a decision threshold other than 0.5 *can* be established and evaluated. The performance criterion used for the evaluation of voice activity detection is the area under the receiver operating characteristic curve (ROC AUC, cf. Section 6.4.3), also used for example in [Eyben and Weninger, 2013]. It considers the entire spectrum of possible decision thresholds.

9.2.2. Results and Discussion

Figure 9.2 visualizes the result of the (outer) cross-validation. Keep in mind that an ideal classifier's ROC curve would go from the lower-left to the top-left corner and then to the top-right corner, covering as much area under the curve as possible (ideally 100% of the graph). It is evident that logistic regression and support vector machines trained using stochastic gradient

descent[1] are comparatively unsuitable for the task. The other classifiers performed surprisingly well and uniformly ($AUC = 0.95$). Only Gaussian naive Bayes fell short with a still impressive score of 0.93.

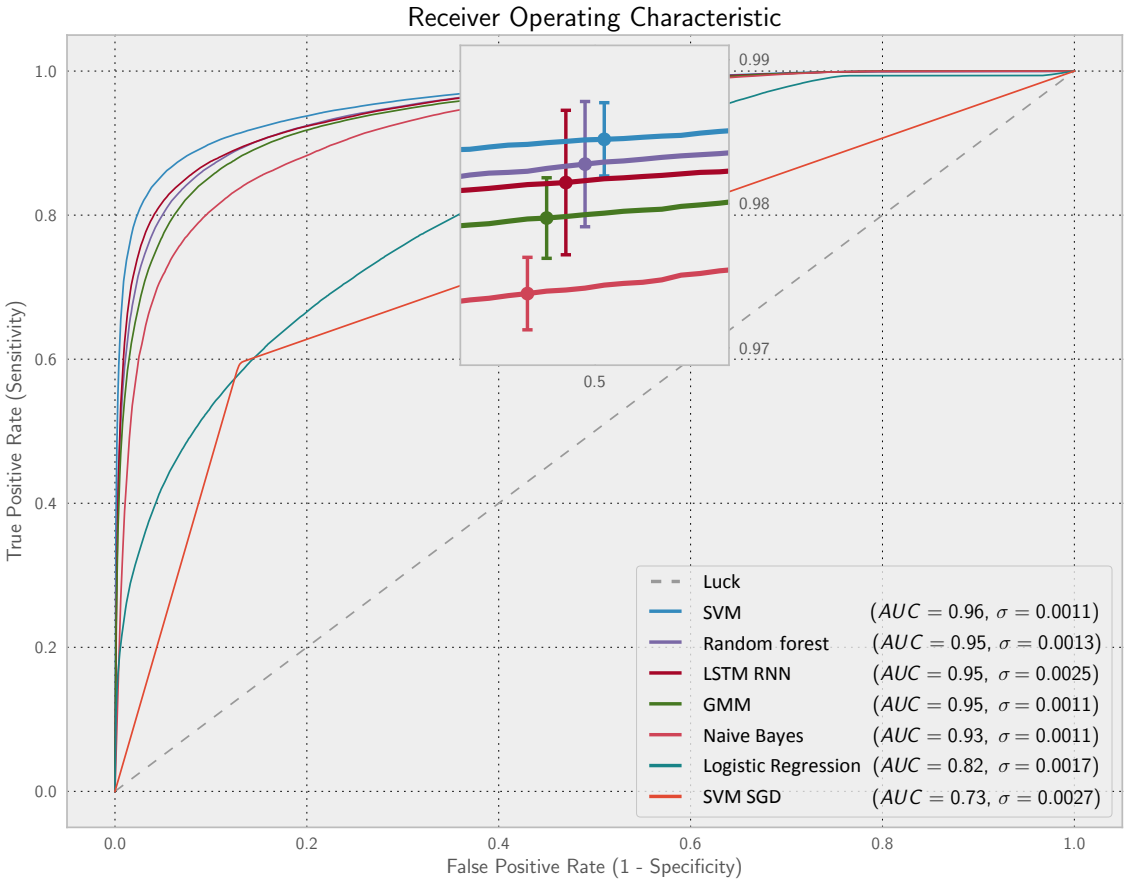

Figure 9.2.: *Receiver operating characteristic (ROC) curves and corresponding area under curve (AUC) of different classifiers for the task of voice activity detection (more is better). σ denotes the standard deviation of AUC among outer cross-validation folds (less is better). Error bars in zoomed-in area show standard deviation of the true positive rate among outer folds at a fixed false positive rate of .5. $N = 200,000$.*

Figure 9.3 depicts the respective confusion matrix of all classifiers as a *four-fold plot*. While the ROC AUC integrates over all decision thresholds, a fixed value had to be picked for calculating the confusion matrices. Decision functions were left unchanged with the default threshold of 0.5 although we calculated an optimal threshold of about 0.39, approximately the same as the rate of positive (voiced) samples in the dataset [2]. The optimal threshold was found by

[1] SVM SGD does not provide classification scores, only categorical labels. This results in a single-point ROC estimate (threshold 0.5) as opposed to a smooth curve.

[2] Further experiments showed little difference in performance for both thresholds.

maximization of the Youden index [Schisterman et al., 2005] and was also roughly the same as the equal error rate, i.e. the ratio of true positive and false positive rate. Counts were summed up over all CV folds.

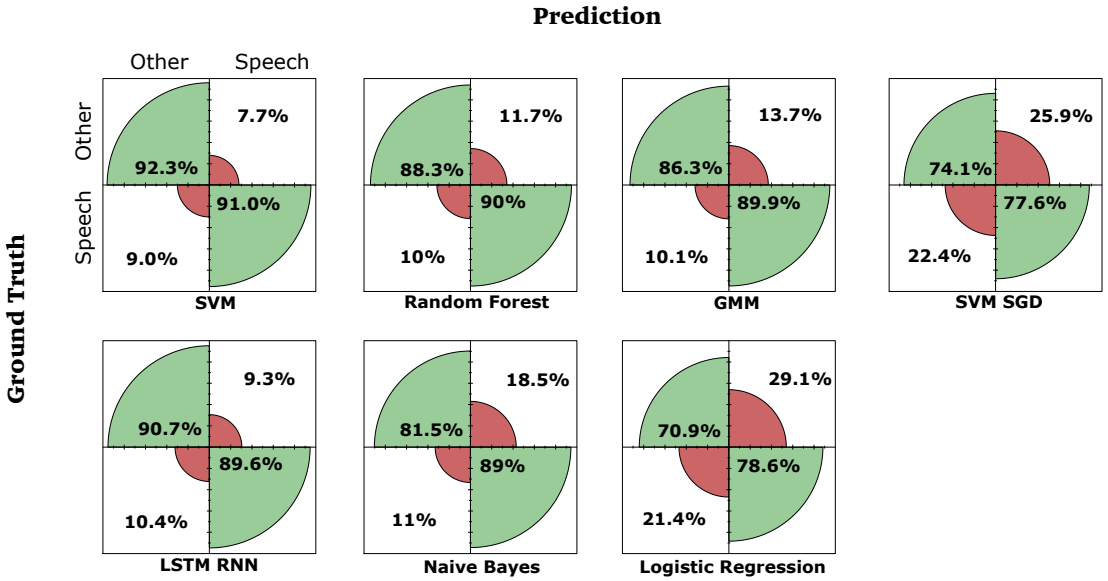

Figure 9.3.: *Confusion matrix of true and predicted classes in voice activity detection accumulated over all CV folds ($N = 200,000$, decision threshold: 0.5)*

All nested cross-validations except for LSTMs exhibited stable model selections and showed no indication of overfitting. For the former, the best performing topology was always single-layered but the optimal number of hidden nodes fluctuated between 100 and 144. Performance discrepancies were negligible. Overall, the following configurations were found to be optimal[3]:

- **Random forest:** $M = 10, \delta = \ln N$, split criterion: entropy
- **Long Short-Term Memory RNN:** $i = 1, H_1 = 144$, learning rate: 1e-2, momentum: 0.9, cost: cross-entropy, optimization: RMSProp
- **Support vector machine:** $C = 1.6$
- **Support vector machine SGD:** $\alpha = 0.001$, learning rate: 1e-07

Furthermore, we statistically analyzed the lists of AUC scores from the 10 outer CV folds as performance seemed similar for some of the classifiers (cf. Fig. 9.2). For neither classifier the null hypothesis that the sample of scores was drawn from a normal distribution could be rejected ($p > 0.05$ using a Shapiro-Wilk test [Falk et al., 2014, p. 105]), thus normality is as-

[3] Classifiers not listed here are non-parametric or used default configurations as described in [Scikit-learn, 2012].

sumed. For any given CV fold the AUC values of all classifiers were closely interrelated as they were computed from the same training and test data. We therefore conducted pairwise one-tailed *paired* t-tests (as opposed to pairwise independent t-tests) [Refaeilzadeh et al., 2009, p. 3].[4] Due to their obvious inefficacy logistic regression and SVM SGD were excluded from statistical analysis. We tested the hypothesis H_0 that two classifiers perform alike at the given task with an undirected alternative (two-tailed test). As this led to $\binom{5}{2} = 10$ pairwise comparisons we adjusted the significance level to $\alpha^* = 0.005$ using the conservative Bonferroni correction [Greene and D'Oliveira, 1985, p. 107]. Detailed results (effect sizes, p values) can be found in Appendix C. All differences were significant except for *Random Forests vs LSTMs*. Subsequently, both will be considered equivalent in terms of performance.

Although there is a statistically significant order over classifier performance (for the given metric and data), it is reasonable to claim that differences among the top classifiers are negligible in practice and that other considerations like computational demands in training and prediction, on-line learning capability, availability of platform-optimized implementations, etc. should dictate the final choice of a classifier for productive use.

Choice of features

We repeated the above experiments with different feature sets:

- Rasta PLP with deltas (18+18)
- Rasta PLP with deltas (18+18), pitch, zero crossings, spectral entropy
- MFCC with deltas (18+18), pitch, zero crossings, spectral entropy

No meaningful differences could be found.

Furthermore, we performed feature selection on our standard feature set (MFCC with deltas) by means of Chi Squared Attribute Evaluation [Setiono, 1995] and Principal Component Analysis [Guo et al., 2002]. Both methods underlined the importance of using the whole set of features. The χ^2 test selected all 36 features, PCA attributed 95% of the variance to the first 32 features. Note that MFCCs are already decorrelated by means of a discrete cosine transform. Normalizing the features with short-term CMVN (cf. Section 6.1) instead of global moments caused significant deterioration (ROC AUC: 0.77). The most dominant distinction between speech and silence lies in the amount of energy in the signal. Short-term normal-

[4] We acknowledge that t-tests are occasionally mentioned to exhibit a "somewhat elevated probability of Type I error" [Dietterich, 1998, p. 1]. However, this is often omitted and alternatives have "not been widely accepted" [Refaeilzadeh et al., 2009, p. 3]. Even Dietterich recommends paired t-tests due to their statistical power if one considers the possibility of missing a difference to be more severe than the chance of erroneously detecting one. We account for the slightly increased risk of committing a type I error by using the conservative Bonferroni correction instead of the equally common but less strict Bonferroni-Holm [Holm, 1979] correction.

ization marginalizes this discrepancy for windows with homogeneous content. We therefore hypothesize that localized normalization doesn't impede the identification of non-stationary ambient noise but impairs the detection of silence.

9.3. Impact of unknown impulse responses

9.3.1. Methodology

To analyze the impact of test data coming from a different recording system than the training data, we conducted two experiments. In the first experiment, **E1**, we simulated an unknown smartphone in a *known* position, in the second, **E2**, an unknown smartphone in an *unknown* position. For each experiment the VAD subset (see above) was convolved once with the training impulse responses and once with the test impulse responses. After that, we randomly sampled 200,000 indices and picked the corresponding frames of each version of the VAD set. Thus, the frames had the same original audio content, differing only in the induced channel effect.

As we had four smartphones, the experiments comprised four evaluations, respectively, over which we averaged. In E1, the training IR set consisted of the twelve IRs of the three non-target smartphones in all four positions. The test IR set represented the same four positions, but of the target smartphone. As an evaluation of all 16 possible phone-position combinations in E2 would have been too time-consuming, we picked the four pairs along the diagonal of the phone-position matrix (cf. Table 9.2). That way, each position and each phone was evaluated once, respectively. Each training IR set consisted of the three non-target phones in any of the three non-target positions.

Phone	**Test position**
LG G2	pocket, microphone upwards
Samsung S3 mini	pocket, microphone downwards
Samsung S1	desk
HTC One X	leather bag

Table 9.2.: *Phone-position combinations for testing in E2*

For this series of experiments we employed Random Forests which constituted the second-best classifier for known impulse responses. Support Vector Machines, the marginal winner (see above), proved to be too time-consuming to train for continuous experimentation. All other parameters including features were the same as in the previous experiments.

9.3.2. Results and Discussion

In E1, the overall ROC AUC score dropped only by a small percentage to 0.94 with the optimal threshold staying almost the same (0.40). Similarly, performance for E2 was 0.93 with an optimal threshold at 0.44. Equal error rates were found to be 14.2 and 15.2, respectively. Confusion matrices for the default threshold of 0.5 are depicted in Figure 9.4.

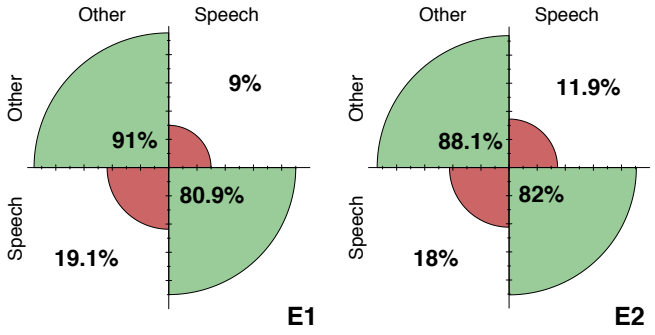

Figure 9.4.: *Confusion matrix of true and predicted classes in voice activity detection with unknown impulse responses – accumulated over all CV folds ($N = 4 \cdot 200,000$).*

When we conducted additional experiments with *no* impulse responses present in the training data at all, performance didn't deteriorate substantially either (ROC AUC: 0.91). However, in that case, the optimal decision threshold soared from 0.39 to 0.7, resulting in bad performance for an unaltered default threshold of 0.5.

In conclusion, the effect of unknown impulse responses on classification performance in voice activity detection can be considered negligible *if accounted for* by the incorporation of *some* channel effects in the VAD model.

10. Speaker Detection

In the preceding chapter, we evaluated common methods for detecting speech in a given signal. In this chapter, we move one step further along our audio processing pipeline to the field of speaker detection. We consider various variations of two common techniques: UBM-GMMs and i-vectors. A third technique, JFA, is ruled out early due to bad performance. We will conclude that particular i-vector configurations with dedicated channel compensation methods perform very well in our use-case (equal error rate < 0.10).

10.1. State-of-the-Art Methods

10.1.1. GMM-MAP

In speaker detection/verification there is only a single target speaker and the set of impostors is open, i.e., unknown. The entire set of speakers other than the target is usually modeled in form of a single so-called Universal Background Model (UBM). The number of components for an individual speaker model is highly dependent on the amount of training data. While some authors, e.g. [Reynolds et al., 2000], list orders from 64 to 256, [Lu et al., 2011] showed that for a small amount of individual training speech, 16 to 32 components are sufficient. For the Universal Background Model the number of components is heavily increased along with the number of training samples in order to represent the broad variety of phonetic classes in the UBM's population. Depending on available computational power and the use case scenario common UBMs range from 128 to 2048 components [Reynolds et al., 2000, p. 26].

In addition to the estimation of the model parameters $\boldsymbol{\theta}$ by the MLE-based EM approach (cf. section 6.2.1), the parameters of an existing model can also be adjusted by MAP adaptation. See [Reynolds et al., 2000] for mathematical details. This allows for adapting the UBM to the target speaker, often by only shifting the means (cf. Fig. 10.1), instead of training a speaker model from scratch. Benefits of this approach are, a.o., faster computation of the speaker model, sufficiency of less individual training data through the incorporation of prior knowledge ("speech [...] in general" [Kinnunen and Li, 2010, p. 10]), and a "tighter coupling" between UBM and target speaker model "which allows for a fast-scoring technique" [Bimbot

et al., 2004, p. 436]. The adapted model may be "better able to handle unseen data as it inherits the modeling power of the underlying UBM" [Togneri and Pullella, 2011, p. 32]. MAP adaptation constitutes the de-facto standard for GMM-based speaker detection systems throughout the literature.

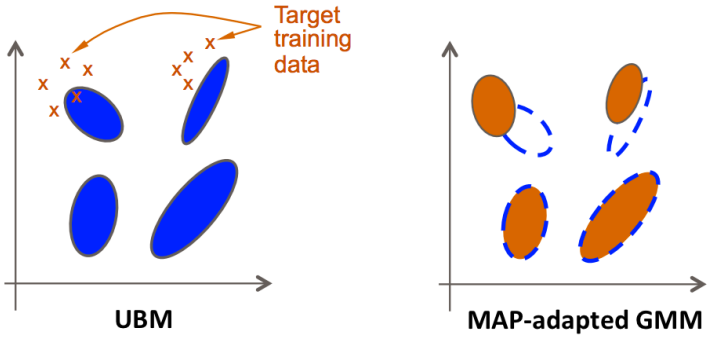

Figure 10.1.: *MAP adaptation (based on [Dehak and Shum, 2011])*

Scoring

We evaluate two common types of scoring feature vectors with GMMs: the *GMM-UBM* approach [Togneri and Pullella, 2011, p. 32] simply compares the log-likelihood ratio of the personal GMM and the UBM with a threshold that incorporates the priors of both classes.

The *z-norm* approach [Bimbot et al., 2004, p. 439] further normalizes the ratio. A pre-determined set of impostors, the *impostor cohort*, is scored against the personal GMM and the UBM as part of the post-training procedure, mean and standard deviation of the log-likelihood ratio scores are stored. When classifying unknown data, the log-likelihood ratio score of the feature vectors is *zero normalized*. Only scores that according to the cumulative distribution function of the standard normal distribution are highly unlikely to stem from the impostor population are accepted. The desired percentage directly dictates the decision threshold.

10.1.2. Joint Factor Analysis

More recent techniques use Gaussian Mixture Models as an extended feature front-end in form of Gaussian supervectors. Instead of scoring utterances against a trained personal GMM, a new GMM is MAP-adapted from a UBM *for each unknown utterance* that is to be scored. Each utterance, however defined in terms of duration, thus marks the training set of a model that is built upon the UBM as prior for “speech in general”. The means of the Gaussians of this new, "temporary" GMM are then concatenated to a so-called supervector that serves as a feature

vector for any other classifier. [Kinnunen and Li, 2010, p. 18]

To compensate for channel effects the Joint Factor Analysis (JFA) technique was proposed. It "considers the variability of a Gaussian supervector as a linear combination of the speaker and channel components" [Kinnunen and Li, 2010, p. 19]. Using standard factor analysis methods, an observed supervector $\mathbf{M}$ can then be decomposed into independent components: $\mathbf{M} = \mathbf{m}+\mathbf{s}+\mathbf{c}$. In this equation $\mathbf{m}$ denotes a speaker-independent speech component, the UBM supervector. Channel variability is modeled by the equation $\mathbf{c} = \mathbf{Ux}$ where $\mathbf{U}$ is a rectangular matrix, whose columns are the *eigenchannels*, and $\mathbf{x}$ is the vector of channel factors [Dehak et al., 2009, p. 1]. The speaker component is modeled by $\mathbf{s} = \mathbf{Vy} + \mathbf{Dz}$. It consists of the rectangular matrix $\mathbf{V}$, its columns being the *eigenvoices*, and the speaker factors $\mathbf{y}$. $\mathbf{Dz}$ marks a residual component. The hyperparameters of this model, $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{D}$, are estimated on large datasets [Kinnunen and Li, 2010, p. 19]. Channel compensation for an utterance is conducted by simply discarding $\mathbf{c}$ and using $\mathbf{s}$ as the personal model. For more information see [Kinnunen and Li, 2010], [Dehak et al., 2009], [Dehak, 2009].

Scoring

For scoring, the matrices $\mathbf{U}$, $\mathbf{V}$ and $\mathbf{D}$ are used to get estimates of $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{z}$ in terms of their posteriors given the observations (utterance to be classified). The log-likelihood score function and its derivation are described in great length in [Kenny et al., 2007] and [Glembek et al., 2009]. All scores are z-normalized (cf. Section 10.1.1).

10.1.3. Total Variability Space (I-Vectors)

A more sophisticated technique based on supervectors and factor analysis is motivated by the observation that channel factors as modeled by the JFA often contain speaker-specific information [Dehak, 2009, p. 86]. Instead of independently estimating the distinct speaker and channel subspaces like in JFA, one can also estimate a so-called total variability space that makes no distinction between speaker and channel factors. Instead of $\mathbf{M} = \mathbf{m}+\mathbf{Vy}+\mathbf{Dz}+\mathbf{Ux}$ the model of an utterance vector becomes $\mathbf{M} = \mathbf{m}+\mathbf{Tw}$ with the low-rank total variability matrix $\mathbf{T}$ and the total factors vector $\mathbf{w}$, commonly referred to as i-vector (intermediate vector) [Dehak et al., 2011, p. 789]. Channel compensation (and also dimensionality reduction) can be performed by applying linear transformations to $\mathbf{w}$, for example Linear Discriminant Analysis (cf. Section 6.3.2) which maximizes inter-speaker variability and minimizes intra-speaker variability. For more information we refer the reader to [Kenny et al., 2014].

Depending on whether the training data for a target speaker (enrollment data) is used as a whole for creating the personal model through MAP-adapting the UBM or whether the data is split into chunks which are used for individual adaptation, the target speaker is represented by a single i-vector or multiple ones (cf. Figure 10.2). The decision is subject to the intended scoring method. Here, we focus on single-vector models.

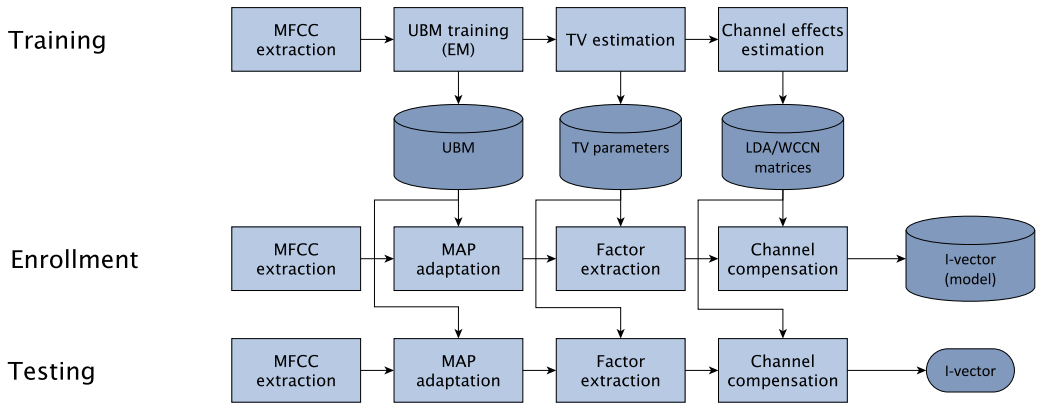

Figure 10.2.: *Architecture and pipeline of a Total Variability system before scoring.*

Scoring

We consider several common scoring procedures. Each is based on one of two meta-approaches to classifying utterances: cosine distance and Gaussian PLDA classification.

Cosine In the first approach, we measure the cosine similarity between the model i-vector and the test i-vector. The resulting score is then normalized. As with GMMs, we evaluate z-norm and disregard t-norm due to its runtime impact. However, as [Dehak et al., 2010] have shown the effect of subsequent zero and test normalization on cosine similarity can be incorporated mathematically into the latter without the need for any kind of explicit normalization. The method relies on pre-computed mean and covariance statistics from a collection of impostor i-vectors. The literature refers to this modified cosine similarity metric as *cosine kernel normalization* [McLaren and Van Leeuwen, 2011], [Shum et al., 2010].

PLDA In the second and most recent approach, the model i-vector is used to train a probabilistic LDA (PLDA) classifier [Prince and Elder, 2007] of the above JFA model

$\mathbf{M} = \mathbf{m} + \mathbf{Vy} + \mathbf{Dz} + \mathbf{Ux}$[1]. PLDAs have been employed successfully in speaker recognition, for example in [Kenny et al., 2013] and [Rajan et al., 2013]. At the time of writing, this approach "achieves state-of-the-art performance" [Rajan et al., 2013, p. 1]. The three common types – standard, simplified and two-covariance PLDAs – are reviewed in [Sizov et al., 2014] and are evaluated as mostly equivalent in terms of predictive power. We treat the PLDA type as a model parameter and consider all three options. All PLDA scores are z-normalized.

10.2. Performance evaluation

We conducted multiple experiments to evaluate the performance of current state-of-the-art methods for speaker detection in our mobile context. It is characterized by the potential presence of a plethora of background noises as well as the likely alteration of acoustic signals through handset variations and fabrics between signal source and receiving device. In Section 9.2 the impact of known and unknown channel effects on performance in voice activity detection was investigated. In speaker detection, contrary to VAD, the suitability of all methods we evaluate for coping with channel effects in general is already assessed by the literature. For speaker detection we therefore focus on unknown channel effects and examine how they are best compensated. All experiments in this chapter simulate the same scenario: a smartphone is carried in an unknown position. It records audio and subsequently performs speaker detection.

The experiment pipeline comprised three steps: training of all general models (UBM, compensation models etc.), user enrollment, i.e. training of the personal models, and evaluation. We particularly focused on a comparison of different enrollment strategies with practical constraints, e.g. minimal user involvement, in mind, using the impulse responses described in Section 7.3. The three enrollment strategies we evaluated were:

E1: Same phone, single position Enrollment data was recorded on the user's target smartphone in a single session and, thus, position. For that we picked the smartphone position with the presumably smallest signal alteration: a desk in front of the user. One might make a point that a more intuitive use-case would be that the user holds her phone in her hand. While we know that signal alterations vary with small positional changes, we expect the magnitude of these alterations for both cases to be very similar as no objects or fabrics lie between sending and receiving end of the signal.

[1] While the total variability is preserved by the i-vector, it is broken down into between- and within-speaker variability by the PLDA. The JFA approach is mimicked in the total variability space instead of the GMM supervector space. Note that it is common to apply both LDA and PLDA subsequently. The former for transformation, the latter for transformation and classification.

E2: Other phones, multiple positions The user's enrollment data was recorded in a single session on a studio microphone under clean conditions. In post-processing, a plethora of positions was simulated by means of convolution with impulse responses of other phones.

E3: Same phone, multiple positions Recording was conducted in multiple smaller sessions (e.g. one minute each) on the user's target smartphone. The position of the phone was different for each session.

10.2.1. Methodology

Experiments were carried out on a custom speaker dataset which contained speech sampled from each of our major corpora: five speakers from TIMIT, Buckeye, Mobio, and Voxforge, respectively (9 female, 11 male). The UBM consisted of speech taken from TIMIT and Buckeye ($\sim 1,500,000$ voiced frames). Voice Activity Detection was performed and only frames classified as speech were processed for speaker detection. We employed Random Forests ($M = 10$) for VAD and used $\sim 5,000,000$ frames of annotated speech from TIMIT and Buckeye for training. All sets were fully disjoint. To improve robustness training data is often recorded in more than one condition (cf. VAD methodology in Section 9.3.2). In the literature this is called multiconditioning [sic] [Rajan et al., 2013], [Kinnunen and Rajan, 2013], [Woo et al., 2006]. Usually, the term condition refers to background noise at a particular signal-to-noise ratio [Ramirez et al., 2007] but can also extend to other aspects like the recording source [McLaren and Van Leeuwen, 2011] and spatial characteristics like reverberation [Garcia-Romero et al., 2012]. For our experiments, we mixed all speech data with noise at various SNR levels and *also* convolved them – distributed equally – with varying impulse responses, both on an utterance basis as described in Section 7.4. This procedure matches what [McLaren and Van Leeuwen, 2011] and [Garcia-Romero et al., 2012] call *pooling* of sources. In its final form, that is, after mixing, the speaker dataset consisted of $\sim 1,000,000$ frames. In terms of features, we kept the design of the VAD experiments described in Section 9.2.1: frames of 20ms length with 10ms offset and 36-dimensional, standardized MFCC vectors including 1st-order deltas.

For the GMM, we selected optimal parameters, i.e. the number of mixtures, using the Bayesian information criterion [Murphy, 2012, p. 163]. The optimal configuration was used later for all supervector based approaches. The scoring method was considered a model parameter as well. Parameter selection was performed in each CV fold, individually. Covariance matrices were diagonal and only means were MAP-adapted. A mixed-gender model was used as "a gender matched UBM performs significantly worse under cross-gender test conditions" [Dunn et al., 2001, p. 2] which we encounter in real-life scenarios.

For all other configurations we picked parameters that are commonly listed in the literature:

- For the plain TV approach with modified and unmodified cosine scoring as well as for the standard PLDA, we used 400-dimensional whitened and length-normalized [Garcia-Romero and Espy-Wilson, 2011] i-vectors. We reduced the number of dimensions to 200 by applying LDA (not to be confused with *PLDA*) followed by WCCN (cf. Section 6.3). The rank of the standard PLDA matrices **V** and **U** was set to 200, respectively [Jiang et al., 2014]. A diagonal covariance matrix was used.
- For the simplified and the two-covariance PLDA 600-dimensional i-vectors were whitened, length-normalized, and transformed by LDA and WCCN. The LDA projection reduced the number of dimensions to 550. The rank of **V** and **U** was set to 300 and 0, respectively. Full precision matrices were used. [Sizov et al., 2014]

For each enrollment strategy E1, E2, and E3 we studied multiple target smartphones and target positions which were not part of the training data. For that purpose the diagonal experiment design described in Section 9.3.1 was reused, leading to four distinct (*smartphone, position*) pairs and, thus, four runs per experiment. Mean performance is reported here. All impulse responses not representing the target phone or the target position (including the approximated ones, cf. Section 7.3.3) were used in the training process.

We averaged performance over all 20 speakers. In each of the 20 iterations per experiment run one speaker was selected as target speaker, the remaining 19 were declared impostors of that iteration. These iteration-specific impostors must not be confused with the general impostor cohort which was used for score normalization. In each iteration we conducted 10-fold cross validation to pick different parts of the data for training and testing for both the target speaker as well as for the impostors. The training subsets for impostors were discarded (for that iteration) since only the target speaker's model had to be trained. As recording enrollment speech is cumbersome for the user, we limited the amount of training speech to a maximum of ten minutes, discarding the rest. In practice, more enrollment data could, for example, be obtained on the fly by tapping the user's phone calls. For TV-based approaches, a single i-vector was computed from the multiconditioned enrollment data which is equivalent to averaging multiple i-vectors of different conditions as is done in [Rajan et al., 2013]. For all of the 20 speakers the test data was cut into chunks of arbitrary duration [Kenny et al., 2013] to simulate realistic turn-taking. The length was picked randomly from the range $[4; 20]$ seconds. Other values that roughly reflect common turn durations would have been equally suitable and remain to be tested. All chunks, i.e. turns, in the test set were shuffled (across speakers), then concatenated to a single big test vector. As the GMM scores frames individually, smoothing over a non-rolling window of 256 frames (2.5 seconds) [Lu et al., 2011], [Xu et al., 2013] was applied, yielding one speaker prediction per window. The same fixed window size was used for scoring with all variable-length TV-based classifiers to ensure a fair comparison. As

performance metric we chose the equal error rate[2]. However, as the test set consisted mostly of impostors ($\sim \frac{9}{10}$) and the data was thus heavily imbalanced in favor of negative samples, we also report the confusion matrix for further analysis by the reader (cf. Appendix D).

The UBM data was grouped according to speakers ($n = 503$) and reused as impostor cohort as well as for estimation of all transformation matrices (TV, LDA, WCCN etc.), as is sometimes done in the literature, e.g. [Finan et al., 1997], [Bimbot et al., 2004], [Dehak et al., 2011].

Our implementation largely relies on the Bob Spear framework [Khoury et al., 2014].

10.2.2. Results and Discussion

Parameter selection for the GMM was stable and the number of Gaussians was set to 256. Although models of higher order are common in the literature, gains are often very small [Nautsch, 2014, p. 88ff.]. Surprisingly, JFA exhibited abysmal performance in most experiment runs. We therefore abandoned it mid-experimentation due to its computational demands.

Figure 10.3 shows a **custom** boxplot of the results. The highlighted center of each box represents the classifier's EER score averaged over all test speakers. In practice, the decision threshold (here: the EER threshold) is chosen a-priori on a test set and remains the same in all use-cases. Hence, we computed a given speaker's EER score over *all* CV folds and positions (runs), not as an average of individual folds or runs. The error bar marks standard deviation of EER scores among speakers while the two single points above and below the bars mark the best and the worst speaker-specific EERs. The box, as a second error indicator, marks the standard deviation among positions of the smartphone.

Best performance was achieved with enrollment strategy E3 and the cosine TV classifier with an average error rate of only 15%. The difference between E3 and E2, however, is small. Only for GMMs E2 seems unfavorable. There are two potential explanations: either TV-based approaches are generally superior to GMMs with respect to speaker recognition. Or the performance gap is due to dedicated compensation methods for channel effects (LDA and WCCN) which are not available for the GMM.

On the other hand, we see a noticeable discrepancy for TV-based approaches between E1 and the other two enrollment strategies. If channel compensation worked properly, it should marginalize position-specific effects and we would observe no meaningful differences between E1 and E3. We will revisit this observation in Section 10.2.3. We hypothesize a) that TV-based approaches are generally more suitable for the task and b) that to unfold their potential these

[2] *here*: ROCCH EER, to be more specific; see [Brummer, 2010, p. 72ff.]

approaches rely on multiconditioning with respect to channel effects. While this is in line with the reported importance of multiconditioned enrollment data in [Rajan et al., 2013], the disparity between clean and multiconditioned data is *far* more gaping in our scenario.

Figure 10.3.: *Custom boxplots of speaker recognition performance (metric: equal error rate) for various enrollment strategies.*

In section 7.3.3 we saw that in terms of magnitude of their spectral alterations smartphone-specific effects are exceeded by position-specific effects. This observation is reflected in the similarity of the two multiconditioned experiments, E2 and E3, for TV and PLDA.

Between-speaker variance seems to be slightly smaller for cosine-based scoring methods than for PLDA while position-specific variance is equal for all approaches. As is to be expected, performance deteriorates slightly the more the smartphone is covered. Analogously, the average optimal decision threshold increases and the identification processes gets stricter.

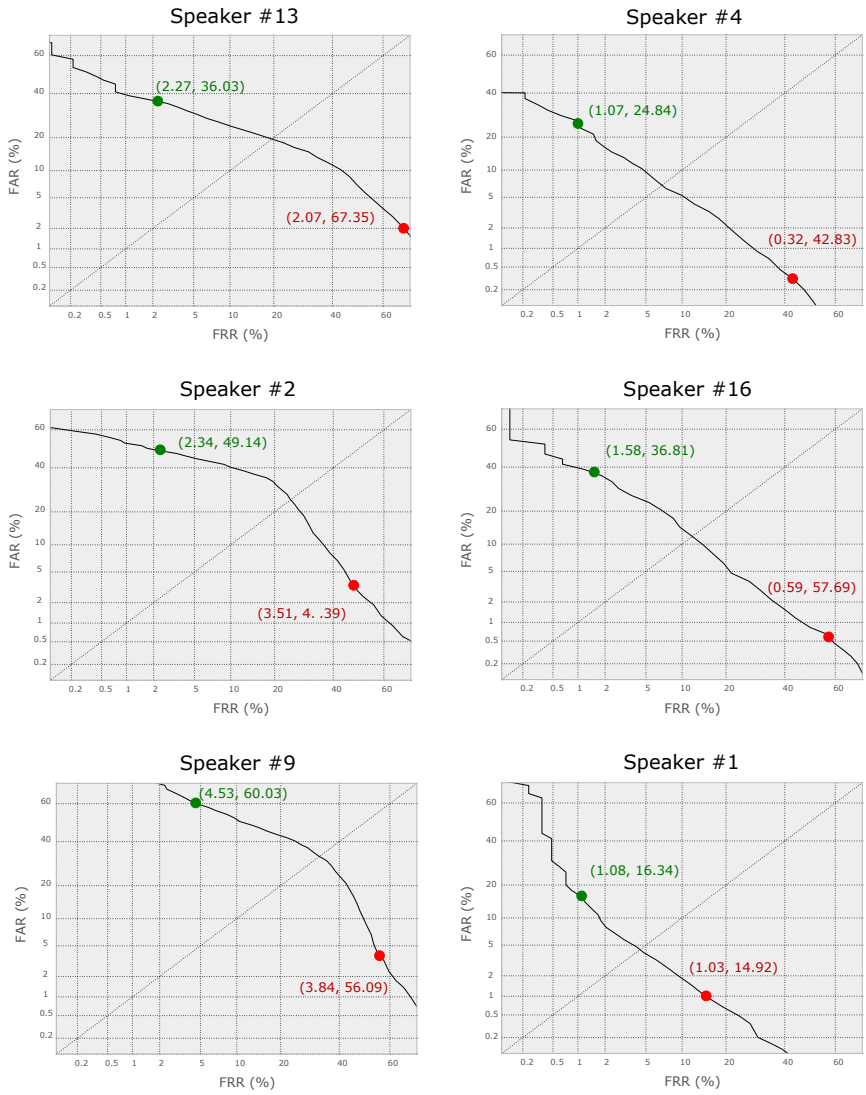

Figure 10.4.: *E3: DET curves of speakers (non-linear axes). Green and red point mark FRR/FAR of maximum and minimum position-specific EER thresholds.*

In practice, the decision threshold is set a priori with the help of a test set[3]. This raises the question: how does the system perform in a position that radically differs from the one that helped determine the threshold? To answer this question the detection error trade-off (DET) curves [Martin et al., 1997] for a randomly drawn subset of the speakers in our evaluation (TV cosine kernel) of E3 were plotted (Figure 10.4). Usually, we set the *overall* EER threshold over all conditions (positions) as decision threshold. Here, we also calculated EER thresholds for each condition individually, selected the maximum and the minimum threshold, and

[3]It is common to either aim for equal error rates or optimize overall performance and accept uneven rates of false acceptance and false rejection.

marked the according false positive rate and false negative rate in the DET plot. The results underline how crucial it is to have data sets that include extreme situations and reflect realistic proportions of conditions. For some speakers, changing the position of the smartphone may otherwise lead to disproportionately high misclassification rates. This can, for example, be observed in the plot of speaker 16: while the overall EER threshold leads to an equal error of roughly 12% (half total error rate: 12%), the EER threshold for an extreme position shifts the error to a false acceptance rate of 1% and a false rejection rate of 58%, thereby massively worsening overall performance (half total error rate: 29.5%). This effect is far greater in speaker recognition than in voice activity detection.

Rasta PLP features instead of MFCCs did not improve performance (E3, TV cosine: EER 0.18, similar DET curvature).

10.2.3. Examination of transformations for channel effect compensation

To revisit and test our hypothesis that LDA and WCCN fail to eliminate position-specific channel effects and that favorable performance is to be attributed to multiconditioning, we reran our experiments for E3 in various configurations with respect to dedicated channel effects compensation methods. If the hypothesis is true, we expect performance to be the same without channel effects compensation. Results are visualized in Figure 10.5.

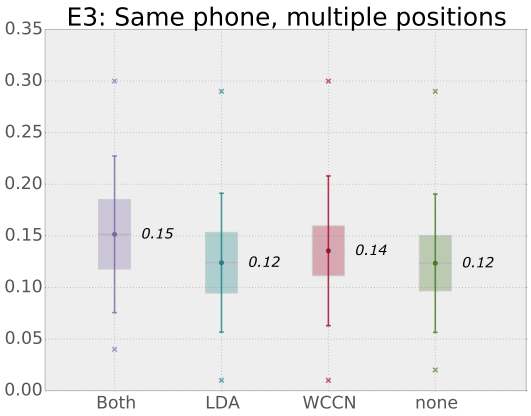

Figure 10.5.: *Comparison of different configurations w.r.t. channel effect compensation. Scores constitute equal error rates.*

It is evident that in our scenario both LDA and WCCN not only fail to improve speaker recognition under mismatched conditions, especially in combination they even exacerbate classification. In its attempt to project the data into a space with minimal variance induced by different smartphone positions, WCCN also discards characteristics of the speaker in the signal. We con-

clude that best results achieved with LDA and without WCCN. This conclusion is not just based on classification accuracy but also on the reduced dimensionality of LDA projected vectors and the resulting speed-up in scoring.

10.2.4. Short-Term Cepstral Mean and Variance Normalization

While in the above experiments all feature vectors were standardized using mean and variance of the training data, we also investigated short-term normalization, often called CMVN (cf. Section 6.1.4), on a by-window basis. CMVN was applied *after* VAD to not impede the detection of silence (cf. Section 9.2.2). In combination with multiconditioning, short-term normalization massively improved performance and reduced the EER for i-vectors with cosine kernel normalization to **0.03**. Previous conclusions held: cosine scoring remained superior to PLDA scoring (EER 0.05) and transformations (LDA, WCCN) showed no impact. Badly picked thresholds still amplified misclassification rates disproportionately.

10.2.5. Variable-length windowing

In speech processing research, the full audio dataset is usually available before an experiment is conducted, i.e., there is no *live* audio stream. Thus, it is common to perform voice activity detection, drop all frames without speech, and further process the remaining frames in subsequently formed windows to smoothen classification results. In a real use-case, however, VAD can't be performed in advance. As windows must only contain speech for speaker recognition to work properly, having windows of fixed length is impractical as available frames aren't processed until the window is filled with speech. During real conversations pauses are bound to occur and processing gets delayed frequently. Variable length also comes with the advantage of an increased probability that the window only contains speech of a single speaker. Thus, higher accuracy is to be expected. But variability in window length comes at a price: with shorter windows one sacrifices reliability in the estimation of the speaker's profile in the i-vector space. In our case, where the smoothing window also constitutes the basis for short-term mean and variance estimation, the drawback also applies to CMV normalization. Along with this trade-off we investigated whether i-vector extraction in the training procedure should match its testing counterpart or whether we should aim for more reliable estimates and make use of the fact that for training all information is available in advance.

Two approaches were compared: in the first, the training procedure for speaker models was left untouched. Non-voiced frames were dropped in advance. Voiced frames were consecutively grouped in windows of 256 and CMV normalized. A single i-vector was computed from the entirety of all voiced enrollment frames. In the second approach, windows of 256

frames were formed prior to voice activity detection. Non-voiced frames were dropped and windows with less than 50% speech discarded. After that, the length of each window lay in the range $[128; 256]$. To match the subsequent testing procedure, i-vectors were extracted on a by-window basis. As we only consider single-vector models, a mean vector was computed in accordance with [Rajan et al., 2013]. Both approaches shared the same testing procedure and i-vectors were extracted like for training in the second approach. No other parameters were changed with respect to previous experiments. Figure 10.6 depicts a plot of the results.

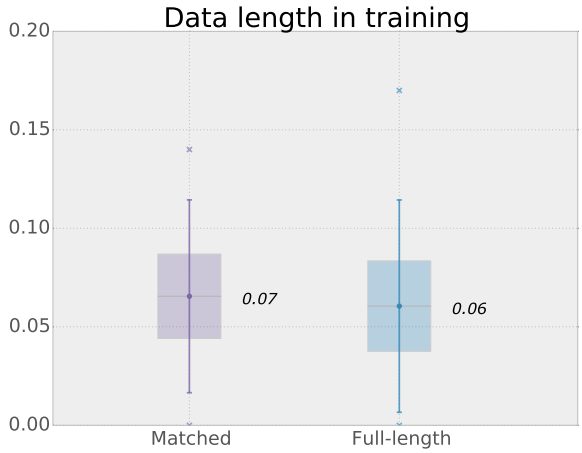

Figure 10.6.: *Data length: should training data be of the same length as testing data or should the maximum be used? Scores mark the equal error rate.*

The on average smaller size for variable-length windows causes performance to slightly deteriorate to an EER of 6%. Nevertheless, the system still achieves impressive accuracy.

Full-length training appeared to be slightly superior to truncated training where the same data length was used as for testing. This finding would be in line with results in [Kanagasundaram et al., 2011]. To find out whether the difference is statistically significant, we conducted further tests. A Kolmogorov-Smirnov test [Lopes, 2011, p. 718] rejected the null hypothesis that either set of scores was sampled from a normal distribution ($p < 0.01$). Instead of a t-test we therefore conducted the nonparametric Mann-Whitney U test [Falk et al., 2014, p. 150]. The null hypothesis of both samples stemming from the same population could not be rejected ($p > 0.05$) and no statistically significant difference was found between the two training approaches.

10.3. Summary

In conclusion, the following statements can be made:

- Speaker detection in a mobile context can be assumed to work reasonably well in everyday scenarios.
- Multiconditioning and CMVN are essential for coping with variance in terms of the position in which the recording smartphone is carried.
- Multiconditioning is also crucial to determine an adequate speaker-specific a-priori decision threshold.
- While both TV based approaches perform similarly, we choose cosine scoring with cosine kernel normalization due to its implicit incorporation of test normalization and due to its fast speed. LDA is performed to reduce the number of dimensions from 400 to 200 and thus to alleviate computational demands.
- With the above configuration, projecting and scoring an utterance of 2.56 seconds takes about 0.06 seconds in our unoptimized implementation on a single 3.7GHz Intel Xeon core. This corresponds to a real-time factor of 0.02.
- Matching the length of training and testing utterances does not seem to influence performance.

11. Affect Recognition

In this chapter we review current state-of-the-art methods for recognizing affect from audio signals. As stated in Chapter 4, continuous affect recognition can be considered a superset of discrete emotion recognition with greater modeling power. The benefits of predicting affective dimensions are twofold: emotions are captured implicitly as well, and we evade the need for training data of particular affective phenomena by instead modeling their underlying affective components.

For reasons of performance we limit ourselves to MFCC features for this task. We will conclude that affect recognition in naturalistic conditions (noise, channel effects) works moderately well and we are able to coarsely capture the four affective dimensions (average correlation coefficient between predicted affect function and ground truth: 0.5).

11.1. State-of-the-Art Methods

Techniques for affect recognition are similar to those used in speaker recognition and modern speech processing in general. Thus, most of the theoretical concepts have been covered in previous chapters. Study of the literature reveals two differences: in affection and emotion recognition, a wider variety of features is used due to the incorporation of prosodic information – this is opposed to a focus on spectral features in speaker recognition. Second, as affect recognition is continuous, classification must obviously be replaced by regression.

Support Vector Regressors (SVR), variations of their classifier counterparts introduced in Section 6.2, have successfully been employed by many researchers [Gunes et al., 2011, p. 829], [Zeng et al., 2009, p. 49] and have also won [Kächele et al., 2014] the 2014 Affect and Depression Recognition Challenge (AVEC 2014) [Valstar et al., 2014]. In [Schäfer, 2015, p. 44] we have achieved results comparable to those of SVRs using Random Forest Regressors if features were limited.

A more recent trend has its roots in the still emerging field of Deep Learning [Han et al., 2014]. In particular, Long Short-Term Memory Recurrent Neural Networks (LSTM, cf. Section 6.2) have been reported to outperform SVRs in many scenarios, e.g. [Wöllmer, 2013, p. 134

ff.], [Eyben et al., 2012], [Tian et al., 2015].

As mentioned above, emotion and affect recognition usually involve a plethora of features. A manageable set of low-levels descriptors if often inflated by the inclusion of delta coefficients and various statistical functionals, with a resulting size in the thousands: e.g., the *Compare feature set*, baseline set for various INTERSPEECH challenges in affect recognition, comprises 6373 features [Schuller et al., 2015]. To reduce complexity, [Eyben et al., 2012] performed correlation-based feature selection on low-level descriptors. For each of the four affective dimensions (Chapter 4), MFCCs are found to be the most important subset: "We see that MFCC are always the most frequently selected features (also supported by Wu et al. [2010b]) [...]" [Eyben et al., 2012, p. 15]. As our current pipeline (voice activity detection and speaker detection) relies only on MFCCs and their 1st-order deltas, we confine ourselves to this limited set of features for affect recognition as well. Compared to sets commonly used in that domain, this cutback appears to be drastic. However, the reader must keep in mind that all feature extraction must eventually be performed on a mobile device in real-time. Restriction to the single most promising set of features seems sensible. Also, some features are simply unsuitable for mobile use-cases. For example, energy/loudness, which correlates strongly with *activation* [Eyben et al., 2012, p. 23], is bound to vary for other reasons than affect: the user might move away from the smartphone or have it covered. Furthermore, fairly successful approaches with feature restrictions similar to ours have been published, e.g. [Eyben et al., 2009], although the evaluation methodology was not comparable to ours.

While most bodies of work perform regression on their features directly, others project them into Total Variability (i-vector) space first, e.g. [Lopez-Otero et al., 2014] who also limited features to MFCCs. The i-vector approach seems more suitable for our use-case as it allows for easy compensation of channel effects, an issue not explicitly considered by those who use MFCCs directly. Lopez et al. advocate the use of i-vectors "due to the ability of this paradigm to get rid of the speaker and channel variabilities". Furthermore, they find that "experimental results obtained in the framework of the AVEC 2013 affect recognition sub-challenge showed an improvement [...] when using the [i-vector] paradigm, as well as a huge dimensionality reduction of the feature vectors used to represent the speech segments. [Lopez-Otero et al., 2014, p. 9] In our evaluation we will compare both approaches.

11.2. Performance evaluation

11.2.1. Methodology

Experiments were conducted using the Semaine corpus (cf. Chapter 7) which contains recordings of naturalistic emotionally colored conversations between an operator and a participant. The former acts according to a predefined emotional stereotype and tries to elicit an emotional reaction from the user. Each recording session is then annotated by a subset of eight specially trained human raters in total with the help of the so-called FeelTrace program. Among other things, it provides adjustable sliders for the four affective dimensions listed in Section 4.1: *valence, activation, power/potency, novelty/expectation*. Each dimension is mapped to the interval [-1;1]. Furthermore, sessions were fully transcribed on the word level.

We partly aligned our methodology with the one described in [Eyben et al., 2012, p. 8 ff.]: As sessions were annotated by multiple raters, we computed average labels as ground truth (cf. distribution in Figure 11.1), omitting sessions with less than three rater annotations. Unfortunately, some of the annotations are malformed with respect to length. Annotations longer than the corresponding sound files were truncated. If any affect annotation was considerably shorter than the sound file (< 99 %), the session was omitted, otherwise truncated. We ended up with 69 sessions of 20 participants.

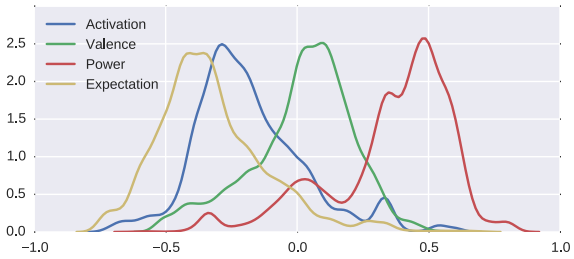

Figure 11.1.: *Affect distribution in Semaine*

The data was split up and grouped by participant. We conducted speaker-independent tests as in real life, systems usually don't have access to speaker- and affect-specific training samples. For performance validation we iterated over speakers: the target speaker's data was used for testing, the data of the remaining speakers for training. With our evaluation we wanted to answer the question of how well affect recognition works in our mobile scenario (limited features, noise, unknown channel effects). Therefore, we mixed Semaine with ambient noise and induced channel effects by convolution of the original signal with according impulse responses. The mixing methodology was the same as the one described in Section 7.4. To each of the speakers we assigned a particular test impulse response. The training data of the other speakers was multi-conditioned on a session basis, i.e. each session was convolved as a whole with an impulse response randomly drawn from the pool of remaining IRs (different position and phone than test data). While we first used the entire set of non-target speakers for train-

ing, we noticed an improvement using subsampling. Hence, we used a random subset of ten speakers for training. All data underwent CMVN (windowed standardization of MFCCs).

Under the assumption that emotions/affect components are rather static within a moderate time frame (socio-emotional real-time, cf. Section 4.1.1) and don't fluctuate wildly, [Eyben et al., 2009, p. 14] apply a low-pass filter on affect predictions to smoothen out high-frequency fluctuations (blatant example of an unwanted time series prediction: 1- 2- 0 - 1 - **40** - 2 - 1). Through cross-validation we evaluated the filter specified in said publication, a Butterworth filter [Parks and Burrus, 1987] as well as a Savitzky-Golay filter [Savitzky and Golay, 1964], the latter two being common low-pass filter implementations (e.g. [Ferris et al., 2005], [Pan et al., 2003], [Ruffin and King, 1999], [Chen et al., 2004]). The filter's optimal window size was determined via CV as well. Note that applying a low-pass filter comes with a trade-off: for a given prediction, half of the window has to be known in advance. A prediction delay is introduced subject to the window size, thus the latter should not exceed a *reasonable* length for socio-emotional real-time recognition.

For the time-sensitive LSTM we employed a strategy once again similar to the one described in [Eyben et al., 2012, p. 18] where sessions were split in sequences. For the sake of simplicity we didn't align sequences with user turns but rather divided each session into 50 sequences to obtain sequences of varying length (across sessions), roughly comparable to those listed in the above body of work. For the NN topology we used a design with only one hidden layer and 50 units, the same design that was used in [Eyben et al., 2009, p. 14] with a similarly reduced set of features. The authors highlighted the importance of adding Gaussian white noise to the training data to prevent overfitting for some configurations. We not only made the same observation, presumably due to the channel effects we even had to increase the standard deviation from $\sigma_n = 0.3$ to $\sigma_n = 0.9$ to get a good generalization. A learning rate of α =1e-6 was picked for gradient decent with 70 mini-batches and a momentum of 0.8 (cf. Section 6.2.6 for optimizer, Currennt implementation). Training was stopped after 100 epochs or after 5 epochs without improvement to the validation set (10% of the training data). It must be mentioned that some trained LSTMs exhibited erratic behavior: despite convergence and low cost evaluations on the validation set, predictions appeared to be random (no correlation to ground truth at all). As this could be checked without involving the test set, training was reiterated in such a case with different initialization in terms of weights and noise.

Parameter selection (classifier parameters, VAD threshold, smoothing window size) was conducted with 10-fold cross validation.

11.2.2. Results and Discussion

As was to be expected, results improved with the width of the smoothing window. The improvement saturated at a width of about 25 seconds. In live use, a window of that size would introduce a delay of 12.5 seconds which we consider a perfectly acceptable trade-off. Best performance was achieved with a Savitzky-Golay filter. The differences between the tested VAD decision thresholds ($0.3, 0.4, 0.5$; cf. Section 9.2.2), i.e. the aggressiveness of the voice detection, were minor.

Result of our tests with a smoothing window of 25s and a VAD threshold of 0.5 are listed in Table 11.1. Unfortunately, the i-vector computation very frequently raised numerical problems using the standard OpenBLAS linear algebra library. Hence, we decided to omit them. In the table below we included results on clean Semaine data reported in recent literature for reference. Note that those results were obtained under different conditions with different methodologies.

Dimension	*LSTM*	*RF*	*SVR*	*SVR Ref I*	*LSTM Ref II*	*LSTM Ref III*
Valence	0.399	0.244	0.280	−0.085	0.454	0.48
Activation	0.570	0.379	0.342	0.653	0.757	0.47
Power	0.509	0.294	0.251	0.367	0.522	–
Expectation	0.517	0.306	0.178	0.190	0.549	–

Table 11.1.: *Affect recognition performance for the Semaine corpus (**with background noise and channel effects**) using only MFCCs. Results reported in the literature (I, II: [Eyben et al., 2012, p. 19, 20]; III: [Eyben et al., 2009, p. 15]) were obtained under different conditions and with different methodologies.*

The above results are ambiguous. On one hand we were able to achieve results that don't deviate tremendously (though visibly) from those reported in the literature – despite our harsh constraints. On the other hand the above figures indicate only moderate performance of state-of-the-art techniques at recognizing affect from speech in real-life situations. Also, a look at the correlation of the affect annotations in Semaine between the various raters (Table 11.2) reveals considerable disagreement across human listeners. This observation has been addressed before by [Eyben et al., 2012, p. 10] who to some extent – but not entirely – question the reliability of the annotation as a ground truth. In any case, it underlines the difficulty of the task in general. The overall correlation between our predictions and the averaged annotations is slightly higher than the correlation of annotations between different raters.

Dimension	Inter-rater correlation
Valence	0.771
Activation	0.567
Power	0.228
Expectation	0.250

Table 11.2.: *Inter-rater correlation for affect annotation in the Semaine corpus [Eyben et al., 2012, p. 10]. The low agreement for power and expectation is striking.*

We conclude that **we are able to coarsely capture the four core affective components** that form human emotions and shape affective phenomena from speech signals in naturalistic environments. We perform best at recognizing activation (CC: 0.57) and worst at recognizing valence (CC: 0.4). The GPU-based implementation of our LSTM runs with a real-time factor of magnitude e-4 on an Nvidia Titan X GPU.

12. A System For Mobile Conversation Analysis and Context Sensing

In this chapter we describe the architecture of a system for capturing conversations on mobile devices and analyzing them with respect to their character. The system employs and integrates techniques presented in previous chapters and implements the processing pipeline as described at the end of Chapter 8.

The system also has a component for collecting additional context information of various kinds for the sole purpose of conducting the user study that we describe in chapter 13.

12.1. Conversation Analysis

In its core the system is composed of two modules: a mobile application for the Android operating system that acts mostly as sensory unit and a remote server application for offloading more demanding computations and for storing information. These two components constitute the *live system* and keep track of ongoing conversations the target user is engaged in in real-time. It is complemented by optional real-time or post-hoc analysis components. The schematics of this architecture are outlined in Figure 12.1.

Using the smartphone's microphone the mobile Android application records audio from the environment of user and device and iteratively feeds the audio stream into a frame buffer. Mel frequency cepstral coefficients (MFCCs, cf. Section 6.1) are extracted from each frame. Consecutive frames are grouped into windows and passed on to a voice activity detection classifier. If the classifier finds more than 50% of the frames in a window to be speech, the window is passed on further, otherwise discarded. Where it is passed depends on whether there is an Internet connection available over Wifi[1]. If so, the window is transmitted to a dedicated server that stores the speaker model (i-vector) and a corresponding threshold (cf. Section 13.1). An i-vector is computed from all voiced frames in the window and compared

[1] One hour of audio surmounts to roughly 100MB of gzip-compressed MFCC data – too much for common 4G data plans in 2015.

to the speaker i-vector. If the score is above the threshold, the speaker is detected. Either way, the window and the score are stored on the server for further analysis. A boolean indicator of the detection result is returned to the client. If no active Wifi connection to the server is available, a local fall-back speaker model (GMM) is used. This model is less complex than the remote i-vector model and therefore less battery-consuming at the cost of accuracy. Using the more complex i-vector models on a mobile device proved to be infeasible due to computational demands. In the fall-back scenario, the window is also stored in the local database.

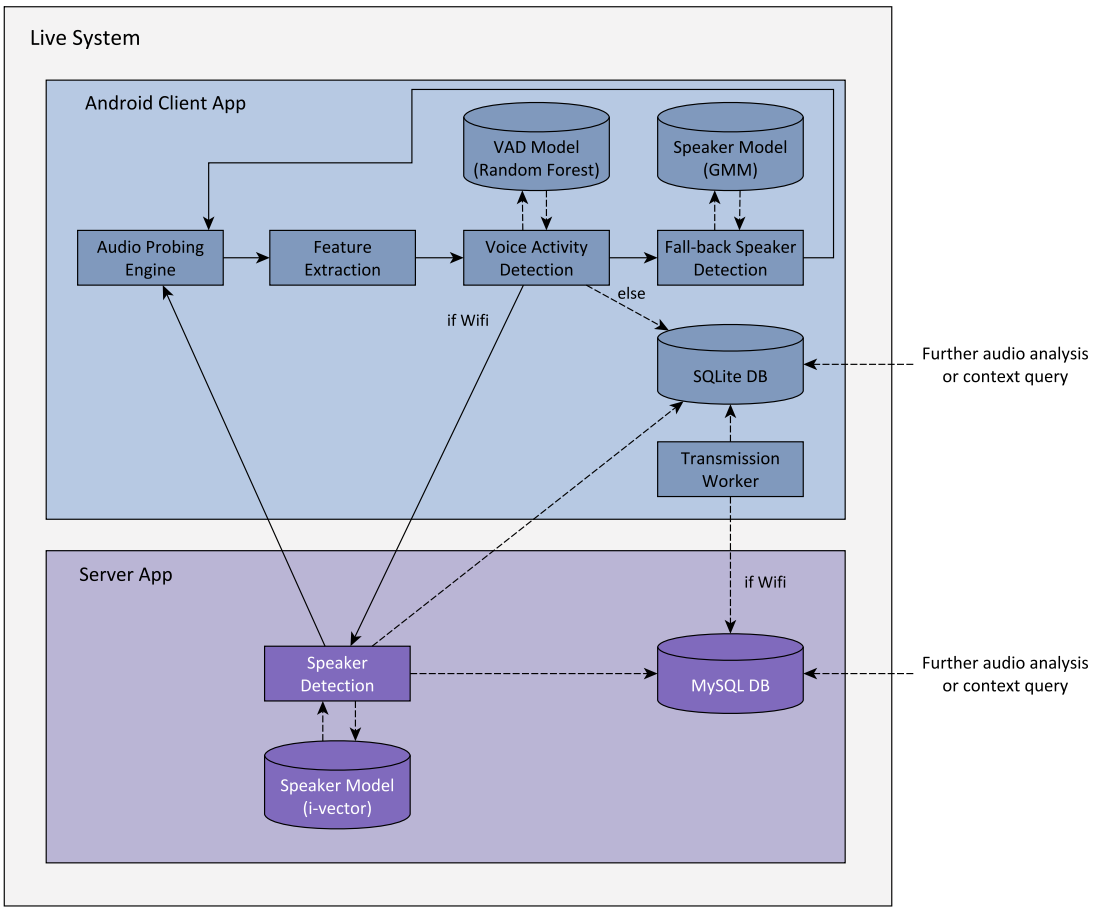

Figure 12.1.: *Architecture of a live system for tracking and analyzing conversations in a mobile scenario.*

Recording on the device is controlled by the so called Audio Probing Engine. To avoid unnecessary computations and battery drainage, it probes periodically for the presence of speech and, thus, conversations. The properties of the probing are dynamic and the engine switches between multiple states. If the presumed likelihood of a conversation involving the target user is low, probing is infrequent and short. If speech and the user's voice are detected, probing is increased in frequency and duration. This leads to constant feedback loops (cf. Figure 12.1).

A worker process running every five minutes on the mobile device keeps both databases in sync whenever Wifi is available. Results stored in any of the two databases can be queried for two purposes: windows with speech can be subjected to further analysis to determine the character of the conversation they belong to and context-sensitive services, like e.g., in our case, a notification management system, can use the database as additional context provider w.r.t. conversations.

In our conversation analysis system we consider only conversations in which the user's voice is detected. Other speech events are treated as random background chatter and are ignored. Thus, the system in general and the Probing Engine in particular rely a.o. on the accuracy of the speaker detection unit which to some degree depends on the presence of Wifi. From that it follows that there is a small theoretical bias towards detecting conversations in settings with Wifi. To minimize it for our planned user study, we take two measures: while real-time conversation analysis on a server is just as feasible as the preceding real-time speaker detection, we conduct our analysis post-hoc. In our study, no live results are required. This allows us to re-run i-vector based speaker detection on the server for those windows that are stored on the mobile device during times when no Wifi is available, even when the speaker is not detected by the local fall-back system. The only remaining dependency on real-time results is with the Probing Engine that has to content itself with the slightly weaker fall-back system. As a second measure, for the study, whenever a new message arrives on the smartphone we automatically override the engine and switch to the most exhaustive probing state. As incoming messages trigger the collection of context information (this will be described in Section 12.3), we can be sure to capture any ongoing conversation from that moment onwards along with all other context information. We ask the reader to bear in mind that even without our countermeasures the bias is a small and temporary one. According to forecasts [Cisco, 2014] of mobile Internet traffic in Germany and extrapolating from a reported annual growth rate of 57% [Ekholm, 2015], by the year 2019 common data plans will be big enough to cover the demands of this system. More importantly: based on mobile CPU performance development [Triggs, 2015] we expect smartphones to be potent enough for local real-time i-vector computation and hence conversation analysis within the next two years (2016/2017).

From a technical standpoint, the details of the various modules are (cf. Section 10.2.1):

Voice Activity Detection Random Forest, 20 trees, $\delta = \ln N$

Speaker Detection 400-dimensional whitened, length-normalized i-vectors; 200-dimensional LDA; cosine kernel normalization; cosine similarity scoring

Speaker Detection (fall-back) 256-dimensional MAP-adapted, z-normalized Gaussian Mixture Models

During the setup of the system, speech samples (*enrollment data*) of the target speaker are recorded on the smartphone (cf. Section 13.1) and sent to the server where both the i-vector model as well as the GMM are built. The trained GMM is then sent back to the smartphone.

To have an optimal speaker recognition decision threshold for scores of unknown speech data that is to be classified, we compute speaker-specific ERR thresholds [Brummer, 2010, p. 72ff.] as described in Section 10.2.1.

For the user study, each participant's enrollment data was split into 80% training data and 20% speaker data for computing the EER threshold. The combined enrollment data of all other participants served as non-speaker (impostor) data w.r.t. this threshold.

12.2. Dynamic Probing

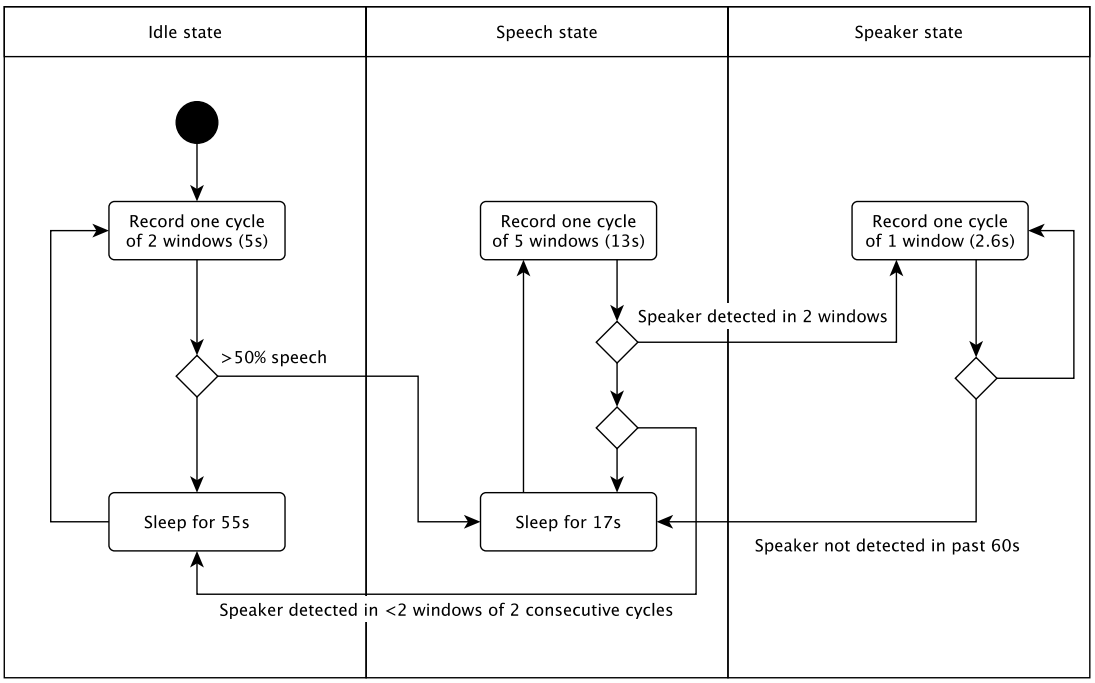

Figure 12.2.: *Activity diagram of dynamic probing with three states (*else *labels omitted).*

The probing engine, responsible for adjusting frequency and duration of any recording, must find a balance between consuming as little energy as possible and not missing any conversation that may emerge. It dynamically switches between three *probing states* subject to evidence and thus confidence that a conversation is taking place (cf. Figure 12.2). In the default state, *idle*, the system is confident that there is no conversation at all. Thus, probing is short

and infrequent. If the system detects speech in the acoustic environment, it switches to the according state, *speech*. At this point, the system assumes a conversation to take place but there is not enough evidence to assume the target user's involvement. Probing is now longer and more frequent. If the user is detected and the system is therefore confident that the user is engaged in a conversation, audio is recorded constantly – even during periods of the other party speaking – until there is no more evidence of the user's involvement. This is to ensure that no short utterances of the user are missed and to allow for unconstrained analysis. State transitions can only occur in incremental steps, no direct jump from *idle* to *speaker* is possible and vice-versa.

Parameters are chosen in accordance with empirical data from the literature: in *idle*, the system listens for any evidence of speech. As the probing is infrequent, it is important to always record a long enough stretch of audio to ensure naturally occurring pauses in a conversation be averaged out. Otherwise the system might by chance capture mainly a pause and thus extrapolate from an instant that is not representative. In general, more than half of any window captured during a conversation must contain speech in order for the conversation to be recognized. To determine the optimal probing length we consider the worst case scenario with respect to pauses. According to both [Weilhammer and Rabold, 2003, p. 3] and [Campione and Véronis, 2002b, p. 2ff.] more than 90% of all pauses in spontaneous speech in German are shorter than 2 seconds. Therefore, the duration of a recording should be four seconds or longer. Due to implementation details, our smallest unit for recording, a *window*, is 2560ms long. Two windows span a suitable time frame of 5.12 seconds. However, it has yet to be shown that this also constitutes an adequate upper bound: as the average length of an utterance, a stretch of speech containing no internal pauses, lies somewhere between 2.7 and 3 seconds [Xu et al., 2013, p. 3], [Jiahong et al., 2006, p. 2ff.], the amount of speech captured in an unfavorable moment (*pause, utterance, pause*) still dominates the recording. We must also take into account the voice activity detection classifier: if the false negative rate (speech misclassified as non-speech) is higher than the false positive rate (non-speech misclassified as speech) or vice versa, our theoretical construct no longer holds. Hence, we ensure equal misclassification rates by using the equal error rate as decision threshold for the VAD classifier.

In probing for involvement of the user in *speech* state, there is little theory to consider. While it intuitively makes sense to take into account average turn lengths, empirical data shows too much variance subject to context like for example the conversation topic [Esposito et al., 2008, p. 110ff.] to construct a meaningful general model. As probing should be more frequent and longer than in *idle*, we pick according values between those in *idle* and *speaker*.

The delay between consecutive probing iterations in *idle* and *speech* is 55s (60-second cycles) and 17s (30-second cycles) respectively.

12.3. Context Sensing

In addition to the conversation analysis components described above, the Android app also comprises a sensing component for non-conversational context. While both components are completely independent of one another, they both provide context information relevant for notification management (cf. Section 2.4.1). Since it is our goal to study the utility of conversational context in notification management, this second component is relevant for our user study and provides the baseline context for our research upon which we want to improve. Thus, we will compare the baseline context alone with a combination of baseline and conversational context. For the baseline context, our app collects data from all relevant hardware and software sensors listed in the literature on context-aware notification management (cf. Section 2.4.1):

- Time
- Calendar events
- Environmental noise level
- Contacts
- Activity (inferred from accelerometer and gyroscope)
- Location
- Usage patterns (times of user interaction)

To infer activity we employ the relatively new (2015) Google Play API for Activity Recognition[2]. It distinguishes between the following self-explanatory activity states: `IN_VEHICLE`, `ON_BICYCLE`, `ON_FOOT`, `STILL`, `TILTING` (picking up the device), and `UNKNOWN`. Given enough sensory evidence, `ON_FOOT` can be rendered more precisely as `RUNNING` or `WALKING`.

For our user study, the collection of context data is triggered whenever a new message is received on the device.

12.4. Preservation of Privacy

While Android's location API provides geographical information in terms of latitude and longitude, raw coordinates are not meaningful input for our classifier. They also constitute sensitive data that we don't want to track. To preserve privacy, all coordinates are only stored locally in a *separate* auxiliary database along with a *cluster id*. If a new pair of coordinates points to a location within a 500 meter range of any previously encountered location, the cluster id

[2] `https://developers.google.com/android/reference/com/google/android/gms/location/ActivityRecognition`, accessed September 6, 2015

of that previous point is used. Otherwise a new cluster id is assigned. After assignment of an old or new cluster id, that id is then stored in the regular database as geographical context information. When the app is uninstalled, the separate database with all coordinates is deleted.

All communication between server and client is encrypted with SSL. For practical use in every-day situations there is no need for the server back-end to run outside user space, i.e. the user can host the server application at her own machine. It is only for our study that we host a centralized back-end in order to acquire data for research purposes.

All audio is transmitted in an obfuscated, information-reduced representation (features instead of raw audio). Nevertheless, we have to mention that there are reports of successful attempts to partially reconstruct speech signal elements from MFCC vectors [Milner and Shao, 2002]. No linguistic analyses are performed, i.e. the system makes no attempt to capture or interpret any content of "what has been said".

12.5. Energy Consumption

Feature extraction constitutes the most resource-demanding task along the processing pipeline on the mobile device. The key step in extracting MFCCs is the transformation from time data to frequency data using the Fast Fourier Transform [Smith, 1997, p. 225ff.], its common implementation being "Cooley-Turkey" [Cooley and Tukey, 1965]. By instead employing the proprietary ARM-optimized FFT library SuperPowered [SuperPowered, 2015] we were able to reduce the overall energy consumption of our system by roughly 30%. We describe our efforts with some in-depth analyses on multiple devices in [Wahib-ul haq, 2015].

Constant app usage including continuous conversation tracking, context sensing and client–server communication reduces battery life on average by about 35%. Without the dynamic probing strategy diminution increases to about 45% of the baseline battery life. Battery drainage is most pronounced on lower-end devices.

Part IV.

Evaluation

13. User Study

13.1. Methodology

To answer our research questions (highlighting indicators on the margin in this chapter) and to evaluate the system presented in Section 12.1, a large-scale user study was conducted. Funding was provided by the German Federal Ministry of Education and Research (Bundesministerium für Bildung und Forschung, Förderkennzeichen 01IS12057). The greater part of this chapter is published as [Schulze and Groh, 2016].

To advertise the study which we titled *"smart notifications"*, a two-week campaign was launched comprising public Facebook posts, print media (posters and flyers) in public places in and around Munich, announcements in various lectures at the Technical University Munich, and word-of-mouth advertising among friends and colleagues. Interested parties were promised a financial compensation of 300 euro for their diligent participation in a 14-day study on intelligent notification management which involved installing an Android app and answering a short questionnaire on arrival of any SMS or WhatsApp message [Church and de Oliveira, 2013]. Furthermore, the study would also include an initial and a concluding survey. Registrations were collected through a website that also offered information on the outline of the study and the data to be collected. Apart from due diligence, two additional requirements were stated: a medium- to high-end Android smartphone running version 4.3 or newer, and frequent internet connectivity over Wifi. Throughout the course of the campaign we acquired 181 registrations. Due to public Facebook posts, registrations arrived from cities all across Germany, not just Munich. From this pool we randomly recruited 57 people (16 female, 41 male), 39 of whom reported to be students, the rest reportedly being full-time employees. Age ranged from 18 to 38 with a median of 25.

Our evaluation started with the initial online survey in which participants were asked to rate their supposed interruptibility for each of the 29 conversation types listed in Section 5.1. With this survey we also intended to get participants acquainted with all elements of the conversation taxonomy from which they would have to pick during the study.

The *smart notifications* app was distributed to participants via email two days prior to the start

of the study. After the installation an on-screen assistant walked users through the recording of two 30-seconds enrollment samples in two positions (cf. Section 10.2.1). Participants were instructed to always ensure that their notification style be set in a way that they would not miss incoming messages unless the social situation imposed restrictions.

Once the actual study had started, the app listened for incoming messages via SMS and WhatsApp. Each such event triggered the collecting of sensor information to capture the user's current context (cf. Section 12.3). It also triggered the display of a three-page popup questionnaire encompassing the following questions/requests:

- "Shortly state your social situation (people, activity, setting etc.)" [Free text field]
- "If you were in a conversation, which label best describes its character?" [Drop-down list]
- "How do you rate your emotions in this situation on the following scales?" [100-point sliders]
  - Negative, unpleasant — positive, pleasant
  - Powerlessness, submissiveness — control, dominance
  - Calmness, thoughtfulness — agitation, activity
  - Familiarity, expectedness — surprise, unpredicted
- "How appropriate would each of the following notification styles have been?' [100-point sliders]'
  - Loud ringing
  - Chime
  - Vibration
  - Optical (LED, text, dialog)

Answers were bundled with corresponding sensor readings and transmitted to our servers if a Wifi connection was available at the time. Otherwise they were stored on the device and scheduled for later transmission. To balance usability and practicality with our need for data, a maximum of three stacked questionnaires was implemented in case of multiple missed messages. Newly spawned questionnaires replaced older ones in the stack and hence always referred to most recent messages. To avoid imprecise or incorrect answers caused by vague memory, popups that remained unanswered for six hours were discarded.

Independently from the above functionality, the app periodically captured audio recordings of the smartphone's environment using the built-in microphone. This "probing" for conversations was conducted with varying duration and frequency subject to computed likelihood as described in Section 12.2. In the process, MFCC features were extracted locally on the device and sent to the server. The server stored the features in a database, computed i-vectors, and performed speaker detection (all cf. Section 12.1). Results were sent back to the client app in

real-time (<1s) as the probing parameters depended on them. Since further characterization of speech with respect to affect and social properties was not instrumental for the *operation* of our study, we deferred all related analyses to the end. The participant-specific decision threshold for speaker recognition was computed after all enrollment data for the study had been collected. All non-target participants were used as impostors for the respective target.

On the server side, each morning at 8am a dedicated application counted the number of answers we had received from each participant in the previous 24 hours. To obtain a sufficient amount of data we sent dummy messages (*"Thank you for participating in this study :)"*) to participants whose answer count was below a pre-defined target of six messages per day. From the difference between actual count and target count we computed according probabilities to send out the desired number of messages distributed randomly over the day.

After two weeks the study ended and participants were informed they could uninstall the app. A final online survey was conducted. Its first question targeted the desired autonomy of an intelligent notification system. Participants were asked to decide whether they would prefer an autonomous system built from a user model, a model-based system with optional rules to be defined by the user, or an entirely rule-based system. The latter would include the option to define no rule at all, thus disregarding the user's context. In the second question participants were asked to pick the type of information with higher value for interacting with an intelligent notification system: the type of conversation (taxonomy) or conversational attributes like speed, emotions etc. In addition, participants were instructed to justify their choice in words. The third and final question aimed at eliciting privacy concerns and had participants rate their consent of a hypothetical notification system, running on *a conversational partner's* phone, that would capture and process the participant's voice. For that, a five-point Likert scale was employed.

13.2. Results

Over the course of the study 7,741 *notification bundles* were sampled, each consisting of sensor readings representing the user's context at the time a message arrived, and answers to the according mini-questionnaire. Three participants who had answered less than 30 questionnaires, were excluded from the study ex post and not paid any compensation. To ensure the validity of our data, we discarded 575 bundles where in the corresponding questionnaire the rating of the situation in four affective dimensions (four sliders) had been conducted in less than three seconds, a duration which we deemed too small for adequate diligence (mean duration including discarded samples: 14.96 seconds). This threshold was chosen as it constituted a local minimum in the histogram that visibly delimited a first peak.

13.2.1. Interruptibility and social type of conversation

Figure 13.1 depicts a pie chart with the distribution of the remaining 7,166 bundles with respect to the type of conversation that took place when a message was received.

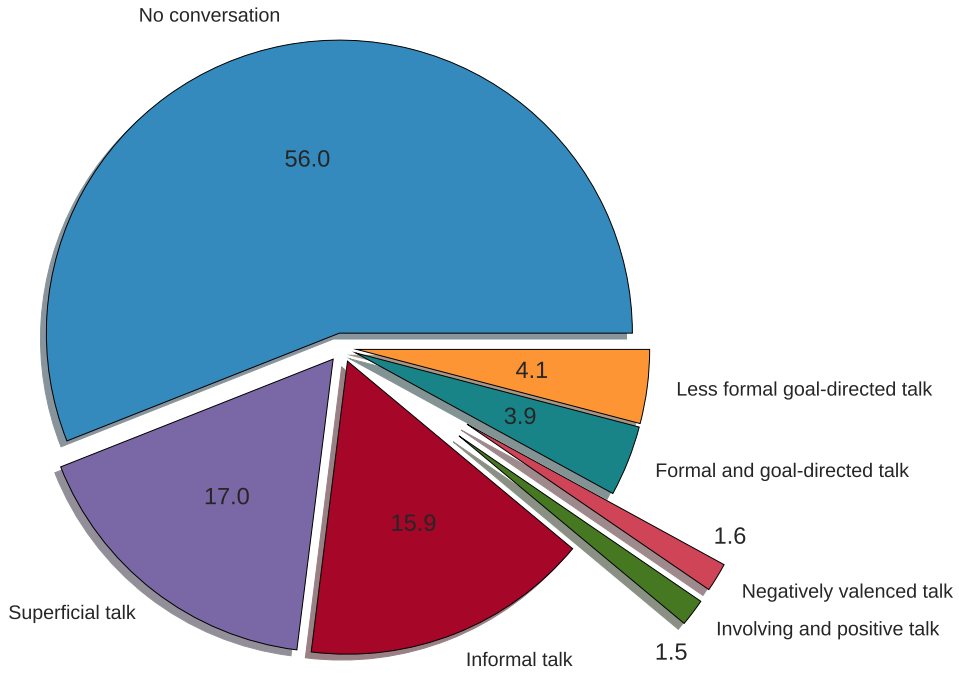

Figure 13.1.: *Relative shares of conversation types in collected data. 44% of the time when a message was received people were engaged in a conversation (based on [Schulze and Groh, 2016]).*

As participants were not asked to state their interruptibility when a message arrived but rather to rate the individual suitability of four notification modalities on a 100-point Likert scale (see above), we inferred an interruptibility score from the invasiveness of the top-rated modality. For that, we ordered the modalities from least invasive to most invasive: Optical (LED, text, dialog), Vibration, Chime, Loud ringing. The index of the modality with the highest suitability score was weighted with said score in percent, resulting in an interruptibility score within the range $[0.0, 4.0]$. If the choice of most suitable modality was ambiguous, the most invasive candidate was picked. We then compared interruptibility scores of different conversation types (cf. Figure 13.2). To equally weigh participants and not attribute more value on the preferences of those participants who were involved in more conversations than the others, only one score per user and conversation type was considered. If a user had multiple interruptibility scores for a given conversation type, the scores were averaged.

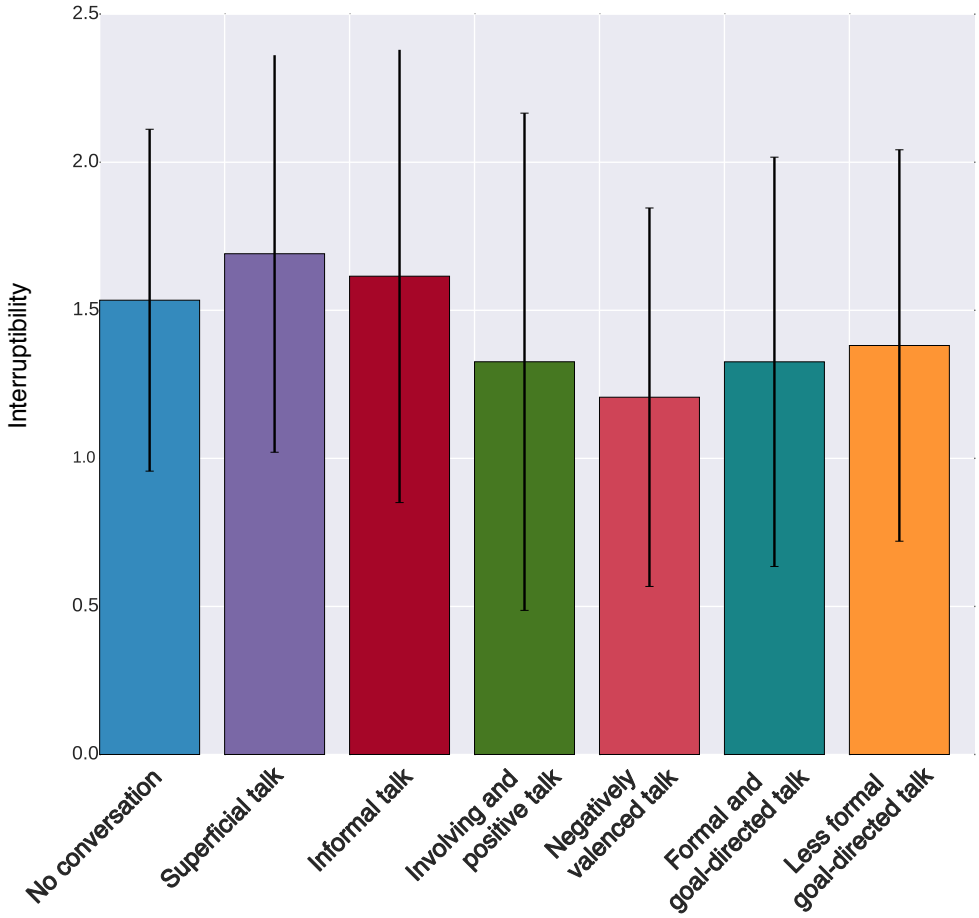

Figure 13.2.: *Conversation types and average associated interruptibility (based on [Schulze and Groh, 2016]).*

It is evident that the various conversation types entailed different receptivities to notifications. **RQ1** Participants were more receptive when engaged in superficial or informal talk than when in other types of conversation, even more receptive than when not engaged in any conversation at all. People were least interruptible when engaged in a negative conversation like "talking about problems", "breaking bad news" or "conflicts". These results indeed reflect those from pre-study presented in Section 5.2. To see whether the differences are statistically significant, we conducted pairwise tests and corrected for multiple comparisons with Bonferroni-Holm [Holm, 1979]. Due to the within-subjects design [Martin, 2004, p. 148] of the study and our previous knowledge from the pre-study we used a one-sided Wilcoxon test for related samples [Falk et al., 2014, p. 150] and considered only participants who had provided interruptibility scores for both types of conversation in question. Note that not all possible pairs were tested, only those that would presumably provide evidence for trends along dimensions as observed in the pre-study. Results were as follows:

Type A	*Type B*	*Q*	*n*	*p*	α^*	$p < \alpha^*$
Superficial talk	Formal goal-directed talk	122.0	37	0.0002	0.0063	✓
Superficial talk	Negatively valenced talk	89.0	33	0.0003	0.0071	✓
Superficial talk	Involving and positive talk	122.0	34	0.0023	0.0083	✓
Informal talk	Negatively valenced talk	129.5	33	0.0035	0.01	✓
Informal talk	Formal goal-directed talk	187.0	35	0.0065	0.0125	✓
Superficial talk	Informal goal-directed talk	201.0	38	0.007	0.0167	✓
Informal talk	Involving and positive talk	214.0	35	0.049	0.025	
Informal talk	Informal goal-directed talk	289.0	39	0.0794	0.05	

RQ1 When plotted against [Goldsmith and Baxter, 1996]'s conversational dimensions (cf. Figure 13.3) the six conversation clusters clearly show a gradual change in associated interruptibility along the depth and to a good extent the formality of a conversation – a change backed up by the above tests: **the more formal a conversation and the higher the perceived involvement of the conversationalists, the smaller the receptiveness to notifications from the mobile device.** The influence of depth is the most pronounced. Valence by itself seems to be *no* indicator of interruptibility.

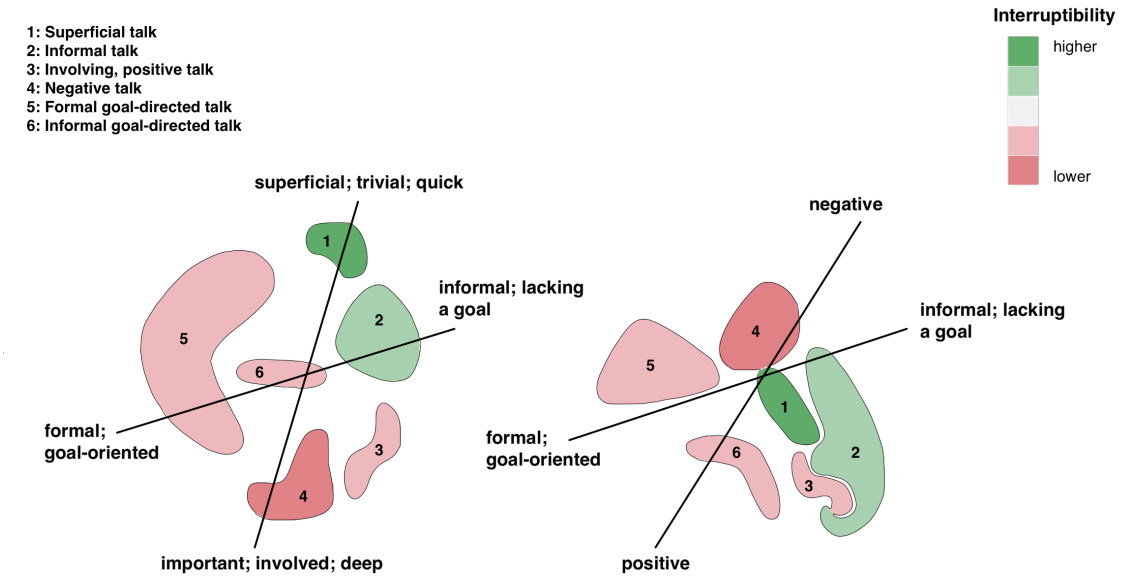

Figure 13.3.: *Heatmap of conversation types and associated interruptibility in [Goldsmith and Baxter, 1996]'s conversation space (based on [Schulze and Groh, 2016]).*

13.2.2. Interruptibility and affective state

Unlike the type of conversation which was represented as a set of discrete categories, the affective characterization of a conversation was continuous. Therefore, two scatter plots are provided to visualize results: one for opportune moments (Figure 13.4), one for inopportune moments (Figure 13.5). Clusters (DBSCAN, $epsilon = 6$, $min = 20$; cf. [Ester et al., 1996]) in the respective affective space are highlighted. Figure 13.6 complements the scatter plots with a histogram of affective states (how often participants exhibited a certain state).

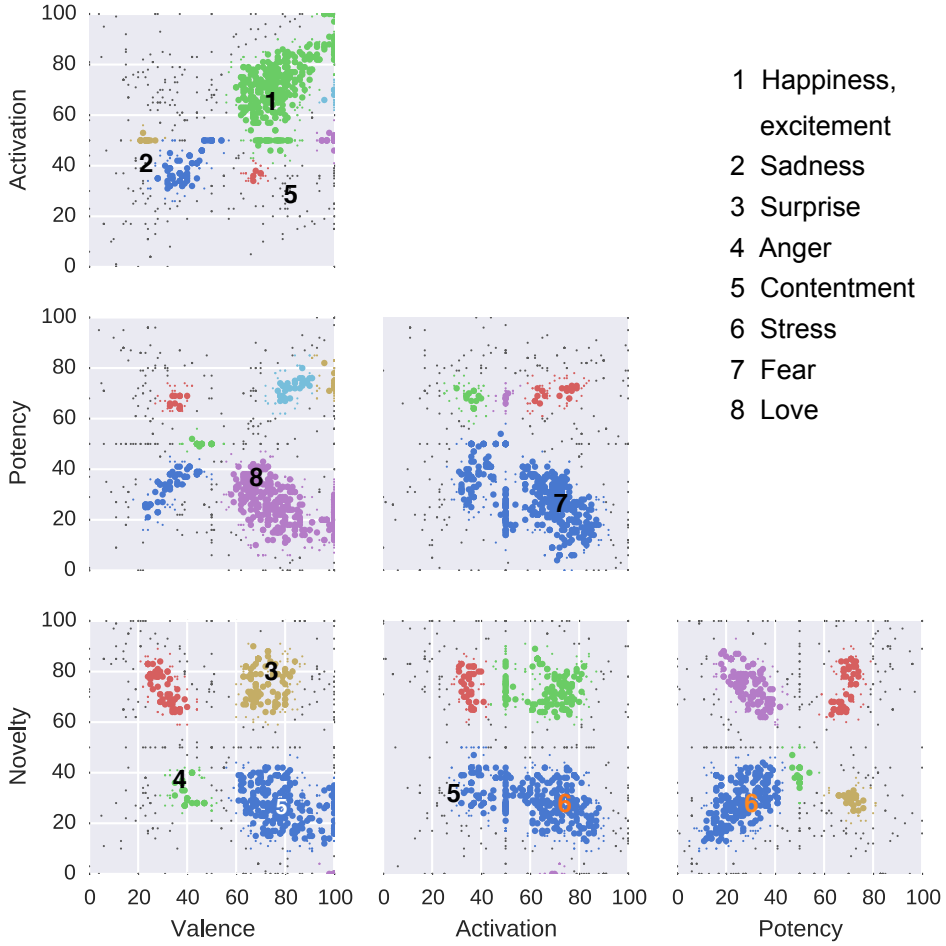

Figure 13.4.: *Scatter plots of* ***opportune*** *moments a notification was received. Each box corresponds to a pair of affective dimensions. The plots show how opportune moments are distributed in the affective space. Clusters are highlighted. Many of them roughly correspond to emotions according to [Fontaine et al., 2007, p. 1055], some important examples are denoted in the plot (indices mark emotion location). The values in each affective dimension range from 0 (negative) to 100 (positive) with 50 corresponding to a neutral state. Based on [Schulze and Groh, 2016].*

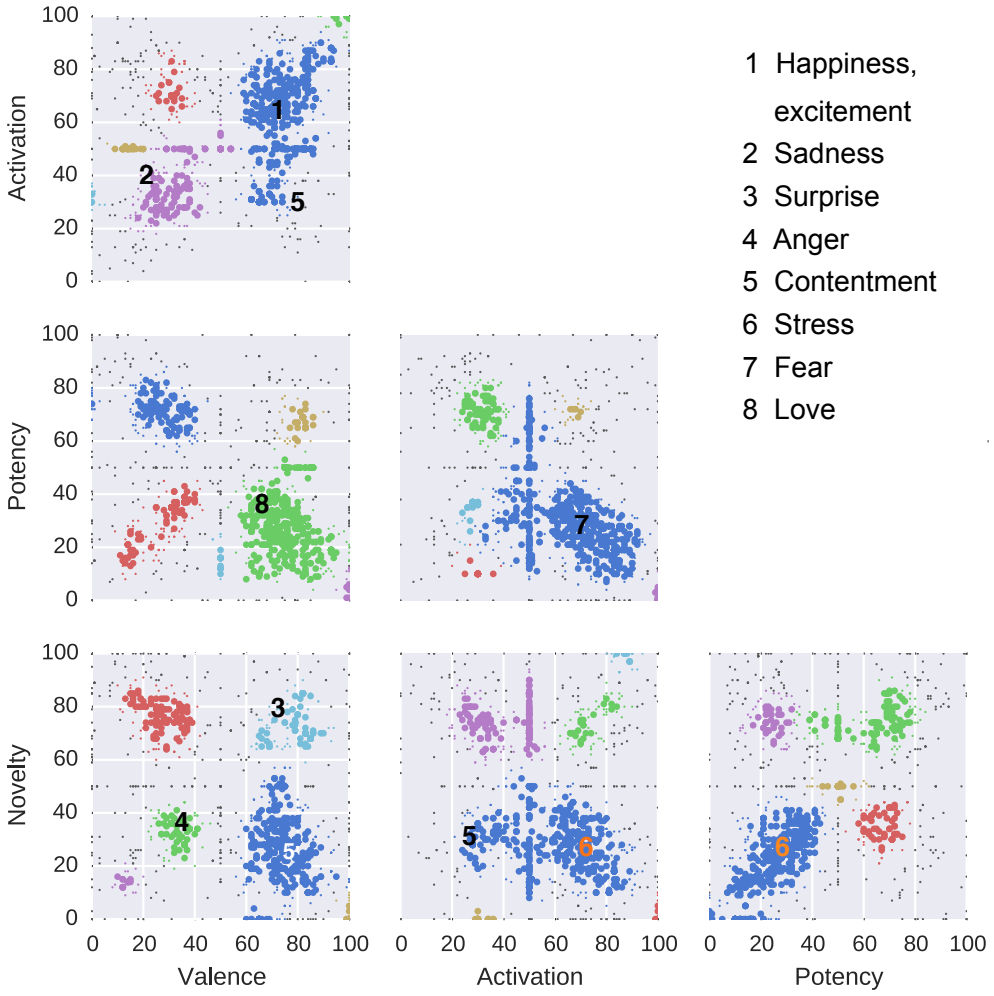

Figure 13.5.: *Scatter plots of* ***inopportune*** *moments a notification was received and how they are distributed in the affective space. Cf. Figure 13.4.*

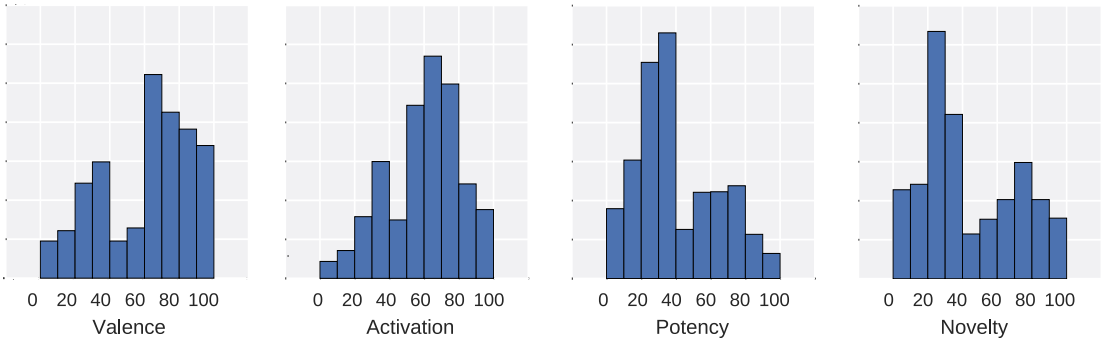

Figure 13.6.: *Histograms of affective states. Participants seem to have experienced many situations that caused moderately high valence ("good") and moderately high activation ("agitated", "aroused"). There are noticeable spikes (many situations) with low potency ("not in control") and low novelty ("familiar").*

The above plots show that moments which are (in)opportune for receiving notifications spread widely over all affective dimensions. However, they do not spread evenly and coarse clusters can be identified. Many samples fall directly on an axis marking a line in the center of the individual images (value of 50 in the respective affective dimension). Other samples cluster around regions in the outer quadrants. The absent continuity and the existence of clusters suggest that interruptibility, high *and* low, corresponds to distinct emotions. The clusters indeed cover or are near regions in which well-identified emotions are located according to [Fontaine et al., 2007, p. 1055], most notably happiness, contentment, anger, and stress. The cluster locations in both figures are similar which implicates that no affective region (emotion) can be *exclusively* associated with a particular interruptibility. The plots don't allow for a closer quantitative evaluation as the cluster sizes in terms of samples cannot be inferred from their width/spread.

To corroborate our belief that there are no trends with respect to interruptibility along any axis, we visualize the *relative* distribution of interruptibility scores for all affective dimensions (Figure 13.7). The dimensions are further divided according to polarity, i.e. whether the samples have a negative coordinate in that dimension (0-40 in the scatter plot), a coordinate near zero (40-60 in the scatter plot), or a positive coordinate (60-100 in the scatter plot). Only potency and novelty show traces of a small trend towards higher interruptibility for higher affective values (diagonal slope) – no definite trend though.

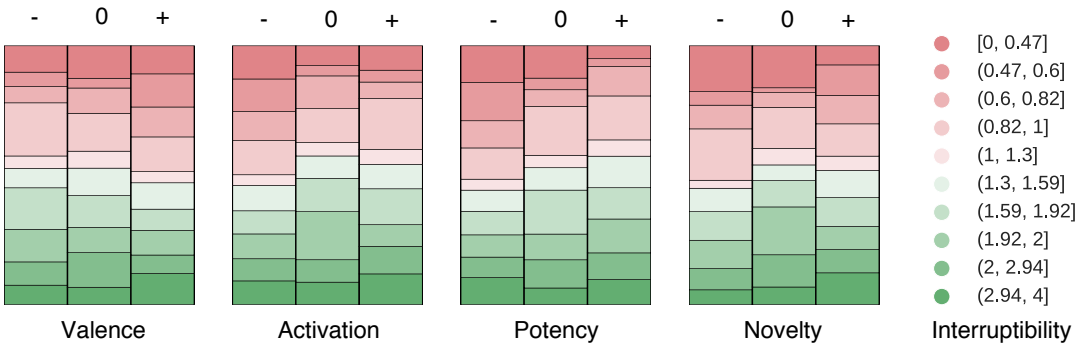

Figure 13.7.: *Relative distribution of interruptibility scores, grouped by negative, neutral, and positive values in the respective affective dimension (based on [Schulze and Groh, 2016]).*

Hereupon, we divided the *valence* × *arousal* space in 3×3 sectors (35-30-35 in each dimension) according to [Russell, 1980]'s circumplex model of affect. Figure 13.8 depicts the model and for each sector the corresponding average interruptibility score (higher values mark more opportune situations). In spite of the obvious differences for the various emotions there is no clear trend along any axis, i.e. dimension, which is in line with Figure 13.7.

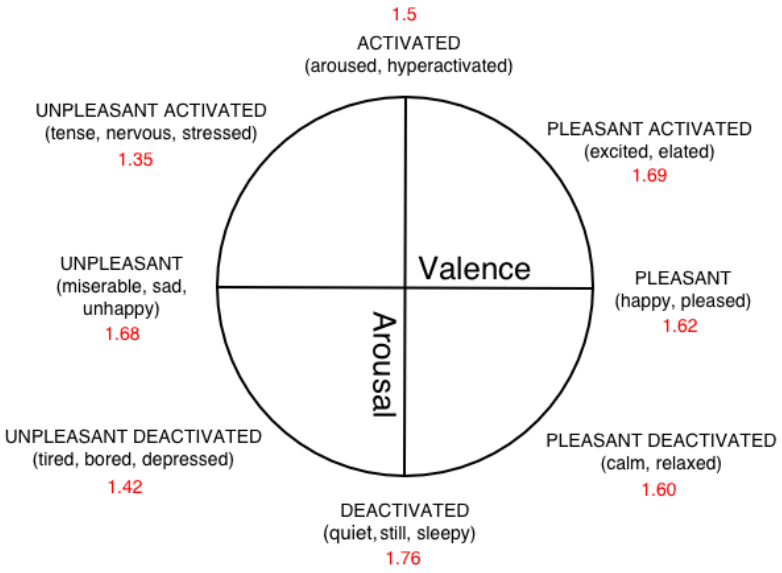

Figure 13.8.: *Circumplex model of affect according to [Russell, 1980] with corresponding interruptibility values.*

Analogous division of the *potency* × *novelty* space yielded similar results (cf. Figure 13.9 to the right). We observe a different value for each sector and a potentially indicated trend along both axes. Compared to *valence* × *arousal*, variance is much smaller here.

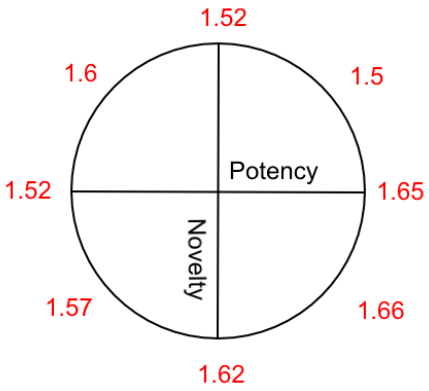

Figure 13.9.: *Interruptibility in the potency × novelty space (cf. [Fontaine et al., 2007] for mapping to emotions)*

In conclusion: none of the four affective dimensions exhibit a *clear* individual correlation with interruptibility. However, the different average interruptibility values of the sectors (Figures 13.8, 13.9, 13.7) for both pairwise combinations and the correspondence of the dominating clusters in each sector to emotions (Figures 13.4, 13.5) indicate that emotions can be associated with certain degrees of interruptibility. This clearly supports the hypothesis that affective characteristics provide utility in notification management. In Section 13.2.5 we will evaluate their potential more thoroughly.

13.2.3. Comparison of social and affective characteristics

We revisit our hypothesis from the introduction of Chapter 5 that an affective and a social characterization of conversations share common aspects but are also intrinsically different. To examine the relationship of both manners of characterization and furthermore to investigate how well social information can be modeled by basic affective dimensions, we employed statistical regression analysis. For that, we first looked at the three social dimensions in [Goldsmith and Baxter, 1996]'s taxonomy: formality/goal-orientation, depth/importance, and valence. For each relevant sample in our dataset[1] we extracted the positional value with respect to the two non-affective dimensions (formality, depth). Unfortunately, [Goldsmith and Baxter, 1996] do not provide a numerical mapping of where in the dimensional space the 29 speech event types lie exactly, they only provide a graphical representation. By placing a [0;100] grid on top of the (rotated) image, we were able to quantify each speech event's position. However, this is only an approximation. We noticed *small* inconsistencies between the two provided plots and computed average values in that case.

13.2.3.1. One-to-one mapping

Having two values per sample for the social dimensions, three values for the affective dimensions (activation, potency, expectation), and one overlapping value that characterizes both (valence) we first computed correlation coefficients between pairs of dimensions. This tells us how well one dimension can be expressed in terms of the other or simply how often comparable values occur. Negative values indicate an opposite direction.

Dimension	*Valence*	*Activation*	*Power*	*Novelty*	*Formality*	*Depth*
Valence	1.0	0.293	−0.248	−0.296	−0.185	−0.190
Activation		1.0	−0.281	−0.170	−0.080	−0.126
Power			1.0	0.268	0.126	0.098
Novelty				1.0	0.162	−0.002
Formality					1.0	0.449

Table 13.1.: *Correlation (Pearson) between pairs of dimensions*

For comparison, we provide the same table calculated from data that underwent more aggressive purging of supposedly suboptimal or even invalid samples. For that, we discarded all samples where the mean edit time was below 12 seconds instead of three. As stated

[1] A sample was considered relevant if the data was obtained during a conversation (44% of all samples).

before, the mean edit time was 14.96 seconds.

Dimension	*Valence*	*Activation*	*Power*	*Novelty*	*Formality*	*Depth*
Valence	1.0	0.273	−0.216	−0.316	−0.118	−0.177
Activation		1.0	−0.349	−0.139	−0.063	−0.089
Power			1.0	0.217	0.1413	0.179
Novelty				1.0	0.258	0.028
Formality					1.0	0.428

Table 13.2.: *Correlation (Pearson) between pairs of dimensions (aggressive filtering of samples)*

According to [Cohen, 1988, p. 77ff.] correlation coefficients, in the social sciences, of .10 are "small", those of .30 are "medium", and those of .50 are "large" in terms of magnitude of effect sizes. These conventions are validated in terms of upper bounds by meta studies listed in [Hemphill, 2003]. If we follow this interpretation, the most pronounced correlations (*valence*/*novelty*, *activation*/*power*, possibly *novelty*/*formality*) are of medium size, the correlation between *formality* and *depth* can almost be considered large. Again, these values indicate how often values in both dimensions co-occur in everyday conversations.

13.2.3.2. Many-to-one mapping

Secondly, we examined a mapping of multiple affective dimensions to single social dimensions and computed regression models in which the target social dimension was expressed as a linear combination of the affective dimensions (*including* interactions between the latter). Full reports can be found in Appendix E.1. The model $formality \sim power * expectation * activity * valence$ (the dependent variable *formality* is expressed as a linear combination of the independent variables, including interactions between any combination of these variables) exhibited a correlation of 24.3 (Pearson's correlation coefficient, cf. Section 6.4.2). The same model for the dependent variable *depth* yielded a correlation coefficient of 24.1. With the more aggressive data filtering mentioned above correlation increased to 40.9 and 33.5. Please note that this doesn't *necessarily* mean that the filtered data is more valid. Our supposedly increased validity comes at the cost of predictive power due to a reduced number of samples. The correlation coefficients suggest that in our data there is indeed an overlap of medium size (see section above) between social and affective aspects of conversations. However, no social dimension can be expressed by a combination of other social or affective dimensions.

We conclude that detection of affective characteristics helps with – but doesn't fully suffice for – capturing the social aspects of conversations.

13.2.4. Inferring conversational characteristics from audio

So far we have looked at our research questions through the lens of social psychology and have only considered the ground truth labels of our data. But how well can the conversational characteristics described by these labels be inferred from audio? How well can information technology capture the user's social and emotional context?

While Chapters 9, 10, and 11 gave us a good notion of the answer to this question, we have yet to examine the performance of speech processing techniques in a real scenario. In our evaluation study we collected audio data of conversations participants have engaged in in moments a notification was received (t_{not}). To be able to handle our analysis, we decided to consider a window of six minutes around t_{not}, three in each direction. That left us with 16GB of audio features corresponding to 1279 notifications from 53 participants. Given the still vast amount of data, we decided to examine the whole six-minute window as a single unit. Note that through this we make the assumption that the character of a short conversation is static. This assumption conveniently aligns with the fact that we have only a single target label for each aspect of the conversation – provided by the user in the on-screen questionnaire. **RQ2**

Dimension	**Correlation**
Valence	0.469
Activation	0.529
Power	0.524
Novelty	0.485
Formality/goal-orientation	0.521
Depth/importance	0.583

Table 13.3.: *Performance for inferring affective and social dimensions from audio obtained through our user study, measured by Pearson correlation between user-stated and predicted values.*

For each six-minute segment we discarded all those windows (256 frames, not to be confused with the smoothing window) that didn't belong to the target user in order to eliminate conversations of other people. From the remaining windows a single i-vector was computed (see previous chapter for parameter listings). This vector represented the user's entire participation in that particular conversation. Vectors (conversations) were grouped by participants. We then built a Random Forest Regressor model (50 trees) for each participant using the entire data set of the remaining participants for training and the participant's own data for testing (speaker-independent analysis). We did not experiment with pre-trained models using the Semaine corpus as a basis as we didn't have enough confidence in its reliability for the reasons stated in Chapter 11. Other models like the top-performing LSTM were tried out but

performed slightly (∼ 5% avg.) worse than the Random Forest. Regression results in terms of correlation coefficients for the various dimensions are listed in Table 13.3 above.

13.2.5. Notification management with conversational characteristics

We have shown that different social and affective properties (sectors in the affective space) entail characteristic tendencies with respect to interruptibility. We have also demonstrated that these properties can be inferred from audio with a reasonable accuracy. This raises the question whether information on conversational characteristics can actually **help improve notification management**. To answer this question we used the notification bundles (sensor data + ground truth via mini-questionnaire) collected in our study to infer the most suitable notification modality based on the user's context. We employed Random Forests with 50 trees ($\delta = \ln N$, cf. Section 6.2.5) to compute participant-specific notification preference models using 10-fold stratified cross validation, and averaged scores over all participants. Participants with less than 30 samples were omitted to ensure a sufficient amount of training data. We compared several models than can be grouped into three categories:

- basic context information **without any conversational characteristics**[2] (minutes since the screen was last turned on, hour of the day, day of the week, minutes to next calendar entry, minutes passed since last calendar entry, number of messages received from sender of the incoming message, location, activity inferred from accelerometer/gyroscope, background noise level in terms of mean absolute amplitude) [**baseline**]

- basic context information plus **conversational context** as provided by the user (**ground truth**):
  - **social type of conversation** in terms of [Goldsmith and Baxter, 1996]'s taxonomy
  - **affective state** of mind characterized in four affective dimensions
  - social type of conversation and affective state

- basic context information plus **conversational context inferred from audio**
  - uninterpreted audio features (i-vector)
  - social type of conversation
  - affective state
  - approximated turn-taking statistics (turn share: $\frac{\#windows_{speaker}}{\#windows_{others}}$, average turn durations in terms of windows)

[2] To guarantee a fair comparison we only considered samples collected during a reported conversation.

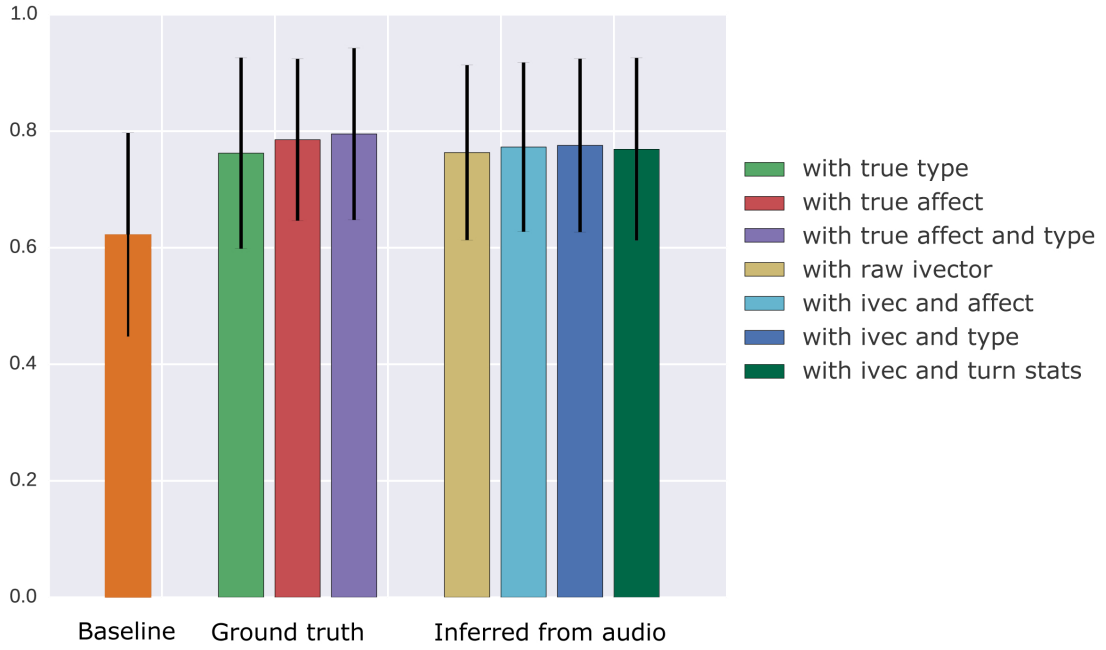

Figure 13.10.: *Classification performance in terms of F1 scores: context (true and inferred) to notification modality (based on [Schulze and Groh, 2016])*

As can be seen in the bar plot in Figure 13.10, **information on conversational characteristics does indeed improve classification** – substantially so. While the F1 score of the baseline, i.e. the basic non-conversational context attributes listed in the literature, averaged 0.62, all classifications that contained any kind of conversational data scored between 0.76 and 0.8. Corresponding confusion matrices can be found in Appendix E. In detail, we make the following observations:

- In notification management, affect (emotion) *seems* to provide slightly greater utility than social properties: 0.785 compared to 0.763 (Wilcoxon's $Q = 60.0$, $p = 0.267$). We emphasize once again the underlying assumption that the affective state of mind of the user is always reflected in her speech. The combination scores 79.5.
- **The uninterpreted i-vector representation of an audio signal carries almost as much information as the ground truth w.r.t. affect and exactly as much as the ground truth w.r.t. social type of conversation** (F1: 0.763). This constitutes a major finding as, from a performance standpoint, it renders further (error-prone) interpretation obsolete. It is a **20% improvement over the baseline**. RQ3,4
- Turn-taking statistics are not reflected in i-vectors. However, they don't seem to provide additional utility (F1: 0.768).

We have stated (cf. Section 2.4) that the context information related to conversation that is taken into account in interruptibility literature is mostly binary: whether there is speech at a given time or not (e.g. [Hashimoto et al., 2013]). In section 13.2.1 we have already shown that this binary indicator is inadequate for determining a person's receptivity to notifications as the latter is subject to the character/type of conversation: e.g. during small talk we are more interruptible than at moments without verbal communication at all (on average). To demonstrate how this finding translates to notification management, we have conducted another classification run with just a binary indicator of an ongoing conversation complementing the baseline context attributes – no other conversational information. As this run used more data than the ones above, which had necessarily included only those notification samples that related to conversations, we have refrained from including it in Figure 13.10. The resulting F1 score of the "binary run" was 0.596 and only marginally higher than a 0.592 run of the baseline context attributes with the same data, suggesting little to no gain from a binary indicator. Further tests do not reveal a statistically significant difference (Wilcoxon's $Q = 141.0$, $n = 24$, $p = 0.80$). As the 0.592 baseline score is near the corresponding baseline score in the above plot (for which we used only part of the data), we consider it safe to state that the binary score can also be compared fairly to the other values of the first runs.

It is evident that **implicit (i-vector) as well as explicit characterization of conversations *considerably* outperforms a binary conversation indicator in automatic notification management**.

13.3. Discussion

As we have seen, implicit characterization of speech events by means of i-vector features yields results almost as good as an explicit characterization in terms of affective coloring or social aspects with respect to a ground truth given by the user. Thus, we have concluded that, from a performance standpoint, explicit characterization as an intermediate step (cf. Figure 13.11) is superfluous. However, classification performance is not the only consideration when designing an automated notification system for practical use as opposed to an academic proof of concept. In our final questionnaire we asked participants how they would want to be supported by a system for notification management. Only 25.93% stated that they would prefer an entirely autonomous system that learned from the user's behavior observed during a training period. The majority of participants (61.11%) expressed a desire to have *optional control* over an otherwise automated system by means of a *set of user-defined rules*. 12.96% would not even want *any* assistance beyond actions defined by rules. This finding is in accordance with the literature that attributes significant importance to a perceived ability of the user to control any system in question (cf. Section 2.4).

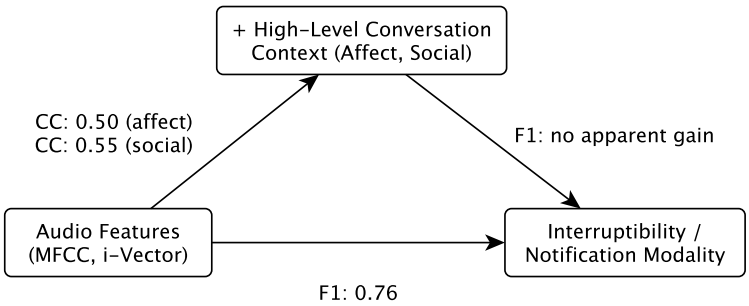

Figure 13.11.: *Classification performance with and without an intermediate step of explicit interpretation of audio (based on [Schulze and Groh, 2016])*

But how can rules with respect to audio be formulated if there is no explicit interpretation of any kind? **From the perspective of usability, well-working high-level interpretation of the conversational context is essential to audio-based notification management.** While significant misclassification in this intermediate step doesn't necessarily hurt overall performance, it renders rules pointless. Given that current state-of-the-art methods perform reasonably well – but not great at the task of high-level interpretation, there is much room for improvement.

From the regression analysis in Section 13.2.3 we have learned that affective and social characterizations of conversations overlap in *approximately* a third of the properties that they measure. This leads us to the question of why a combination of the information both characterizations provide doesn't result in summation of the individual performance gains over the baseline in notification management. Affective characteristics boosted performance by 16.5 points, social characteristics by 14.3 points (both ground truth). Yet their combination makes for an increase of "only" 17.5 points. One potential explanation leads us back to our assumption that the affective state of mind of the speaker, to which our ground truth pertains, is reflected in her speech. While a discrepancy might indeed be the case to some extent, a massive inconsistency would most likely result in abysmal affect recognition performance – which is not the case. Another, much more simple and plausible explanation would be that the overlap translates differently to notification management. If information that can be harnessed in notification management were mostly found in that overlap, then the gain in performance would be accordingly smaller than the sum of individual gains. Consider the following fictional illustration: the social dimension *formality* and the affective dimension *novelty* have a correlation of 0.258 as most of the correlation is found for positive values of both dimensions: formal situations that we are unfamiliar with. These are situations in which we have to conform to norms and don't want to make a bad impression by the ringing of our phone. Therefore, the information of high formality and high novelty is of great utility

in interruption management. In other situations, where there is little correlation, neither dimension is a good predictor for receptivity.

Another important aspect that we haven't treated holistically so far is privacy concerns when capturing audio, speech in particular. While we have covered this point from a technical perspective and have designed our system with the requirement of preserving privacy in mind, the success of such a system in practice stands and falls with how the user perceives its handling of sensible information. In our user study we put emphasis on informing participants how their data would be collected and analyzed, and what measures we had taken to preserve their privacy. After the study, 9.26% stated that they would not feel comfortable if the system ran on the mobile device of *another* person that was part of a conversation the user was engaged in. A further 33.33% felt slightly uneasy about the scenario of their voice being captured on someone else's device. Only 57.41% felt indifferent or had a positive attitude towards the scenario. We reckon that concerns of participants that go beyond our measures to preserve privacy and beyond a lack of their being informed about those measures, can't be addressed by design. A significant part of the population simply does not constitute a potential user base for a partly audio-based notification management system. **Hence, the benefits of our approach over those listed in the literature are not universal, our system has clear limitations in terms of user acceptance.**

14. Conclusion and Outlook

In this final chapter we first want to give a short and concise summary of our accomplishments by directly answering the research questions stated in the introduction. A follow-up section will then shed light on limitations of our system and how we could have approached the matter differently. We conclude this thesis with a short outlook.

14.1. Summary of accomplishments

In this thesis we have evaluated the utility of conversational context for mobile notification management. Motivated by our hypothesis that people's interruptibility in conversations is subject to the character of that conversation we have first looked at different ways for characterizing speech events. As our hypothesis was sparked by intuitive examples of highly emotional situations, affect as the underlying component model was an obvious choice. The second manner of characterization, partly rooted in the affective space as well, was a social taxonomy of speech from the domain of communication research. While it was evident that both approaches overlap, the extent of that overlap was initially unclear and – for our data – later revealed to lie between 25 and 40%. The data was obtained through a large-scale user study ($n = 53$) over the course of two weeks and is therefore considered a sample with high representative power and validity. With its help we were able to prove our initial hypothesis and show that conversations highly vary in their associated average interruptibility depending on their affective and social character (RQ1).

The suitability of state-of-the-art methods from voice activity detection, speaker detection, and affect recognition for inferring conversational characteristics "in the wild" was verified. In the course of our extensive analysis we found that channel effects (signal alterations) caused by the position and manner in which a smartphone is carried are unpredictable and much more substantial than those induced by differences in the device itself. Dedicated transformations and multiconditioning during model training however clearly help alleviate the effects. In real-life scenarios, social and affective properties can be derived from audio recorded by the smartphone with moderate success (regression: correlation coefficient slightly above 0.5, RQ2).

Conversational context in audio was demonstrated to be highly beneficial for predicting suitable notification modalities in automatic notification management. Both affective and social characteristics provide similar utility and either kind improves the performance of personal preference models by more than 20% compared to the baseline set of common context attributes (F1 score: 0.62 → 0.76, RQ3). Explicit decoding of conversational context embedded in speech, i.e. inference of social and affective properties, offers no apparent advantage over the implicit, subsymbolic information of the same kind provided by the raw i-vector representation. However, it allows for the formulation of user-defined rules which constitutes a crucial aspect of usability (RQ4).

14.2. Limitations, critical reflections and outlook

In some of the sections we have already pointed out small limitations of our methodology, for example imperfections of our dataset synthesis, the omitted consideration of temporal dependencies in voice activity detection, the limited user acceptance of audio processing on account of privacy concerns, or the restrictions for creating a rule-based system due to suboptimal inference of context. Even under critical examination, none of these limitations were incisive enough to compromise the validity of our research to any extent. In this section, we therefore want to regard the bigger picture and reflect critically upon our overall approach to improving notification management with the help of conversational context.

As source for conversational context we employed acoustic features without discussing the obvious alternative: linguistic content, i.e. spoken words and their meaning. Additional incorporation of linguistic features, e.g. by means of N-Grams or vector space models, has been shown to improve performance in affect recognition. [Wöllmer, 2013, p. 143ff.] Of course the degree of improvement is highly subject to the considered keywords and the correctness and unambiguousness of their associated affective state. Our decision to focus only on acoustic features was mainly motivated by the additional complexity of linguistic models and their integration, in particular for frame-wise classification (words span many consecutive frames → additional time-dependent models that need to be fused with acoustic models). Also, while there is a small set of established keywords that are known to correlate with affect, a lot of research remains yet to be conducted to optimize it. Nevertheless, we think that the proper consideration of linguistic aspects could greatly increase the accuracy in affect recognition and also in inferring social characteristics of conversations.

Despite decades of fruitful research with respect to interruptibility and context-sensitive systems, intelligent assistants in mobile notification management haven't emerged as mainstream products so far and the transition of evidently well-working systems from academia to the real world remains yet to be seen. Nevertheless, with Wearable Computing becoming

more pervasive there are convincing arguments that research on intelligent notification management will be forced to undergo a revival and that context-sensitivity will find its way into commercial applications. On one hand, wearable devices, as the name implies, are bound to increase pervasiveness of technology in our daily lives and consequently disruptiveness for untimely notifications. On the other hand, it can be assumed that their limited interaction capabilities will render speech the main input modality (cf. Google Glass). Tapping into an already existing speech recognition pipeline with shared audio features will not only minimize the computational impact of our proposed conversation analysis, it will also guarantee the user's consent to the processing of her voice.

We conclude this thesis with the prospect that our findings will help improve notification management on future mobile devices.

Bibliography

[Adamczyk and Bailey, 2004] Adamczyk, P. D. and Bailey, B. P. (2004). If not now, when? In *Proceedings of the 2004 conference on Human factors in computing systems - CHI '04*, volume 6, pages 271–278, New York, New York, USA. ACM Press.

[Amaral et al., 2013] Amaral, T., Silva, L. M., Alexandre, L. a., Kandaswamy, C., Santos, J. M., and de Sa, J. M. (2013). Using Different Cost Functions to Train Stacked Auto-Encoders. *2013 12th Mexican International Conference on Artificial Intelligence*, pages 114–120.

[Anjos et al., 2012] Anjos, A., Shafey, L. E., Wallace, R., Günther, M., and Mccool, C. (2012). Bob: A Free Signal Processing and Machine Learning Toolbox for Researchers Categories and Subject Descriptors. In *20th ACM Conference on Multimedia Systems (ACMMM)*. ACM Press.

[Aronoff and Rees-Miller, 2003] Aronoff, M. and Rees-Miller, J. (2003). *The Handbook of Linguistics*. John Wiley & Sons.

[Atterer et al., 2008] Atterer, M., Baumann, T., and Schlangen, D. (2008). Towards Incremental End-of-Utterance Detection in Dialogue Systems. *Coling 2008: Companion volume: Posters*, (August):11–14.

[Avrahami et al., 2007] Avrahami, D., Fogarty, J., and Hudson, S. E. (2007). Biases in human estimation of interruptibility. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*, page 50, New York, New York, USA. ACM Press.

[Bailey et al., 2000] Bailey, B., Konstan, J., and Carlis, J. (2000). Measuring the effects of interruptions on task performance in the user interface. In *SMC 2000 Conference Proceedings. 2000 IEEE International Conference on Systems, Man and Cybernetics. 'Cybernetics Evolving to Systems, Humans, Organizations, and their Complex Interactions' (Cat. No.00CH37166)*, volume 2, pages 757–762. IEEE.

[Bailey et al., 2001] Bailey, B., Konstan, J., and Carlis, J. (2001). The effects of interruptions on task performance, annoyance, and anxiety in the user interface. In *Proceedings of INTERACT '01*, pages 593–601. IOS Press.

[Bailey and Konstan, 2006] Bailey, B. P. and Konstan, J. a. (2006). On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state. *Computers in Human Behavior*, 22(4):685–708.

[Bastien et al., 2012] Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I., Bergeron, A., Bouchard, N., Warde-Farley, D., and Bengio, Y. (2012). Theano: new features and speed improvements. *arXiv preprint arXiv: ...*, pages 1–10.

[Beigi, 2011] Beigi, H. (2011). *Fundamentals of Speaker Recognition*. Springer US, Boston, MA.

[Bernsen et al., 2012] Bernsen, N. O., Dybkjaer, H., and Dybkjaer, L. (2012). *Designing Interactive Speech Systems: From First Ideas to User Testing*. Springer Science & Business Media.

[Bimbot et al., 2004] Bimbot, F., Bonastre, J., and Fredouille, C. (2004). A Tutorial on Text-Independent Speaker Verification. *EURASIP journal on Applied Signal Processing*, pages 430–451.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc.

[Blackman and Tukey, 1959] Blackman, R. B. and Tukey, J. W. (1959). Particular Pairs of Windows. In *The Measurement of Power Spectra, From the Point of View of Communications Engineering*, pages 98–99. Dover.

[Boil and Hansen, 2010] Boil, H. and Hansen, J. H. L. (2010). Unsupervised equalization of lombard effect for speech recognition in noisy adverse environments. *IEEE Transactions on Audio, Speech and Language Processing*, 18(6):1379–1393.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

[Brummer, 2010] Brummer, N. (2010). *Measuring, refining and calibrating speaker and language information extracted from speech*. PhD thesis, University of Stellenbosch.

[Calvo and D'Mello, 2010] Calvo, R. a. and D'Mello, S. (2010). Affect detection: An interdisciplinary review of models, methods, and their applications. *IEEE Transactions on Affective Computing*, 1(1):18–37.

[Campione and Véronis, 2002a] Campione, E. and Véronis, J. (2002a). A large-scale multilingual study of pause duration. *Speech Prosody 2002. First International Conference on Speech Prosody*.

[Campione and Véronis, 2002b] Campione, E. and Véronis, J. (2002b). A large-scale multilingual study of silent pause duration. In *Proceedings of the Speech Prosody 2002 Conference*, pages 199–202.

[Cellier and Eyrolle, 1992] Cellier, J.-M. and Eyrolle, H. (1992). Interference between switched tasks. *Ergonomics*, 35(1):25–36.

[Chang and Lin, 2011] Chang, C. and Lin, C. (2011). LIBSVM: A Library for Support Vector Machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3):27:1—-27:27.

[Chen and Black, 2008] Chen, H. and Black, J. P. (2008). A Quantitative Approach to Non-Intrusive Computing. In *Proceedings of the 5th International ICST Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, pages 4–5. ICST.

[Chen et al., 2004] Chen, J., Jönsson, P., Tamura, M., Gu, Z., Matsushita, B., and Eklundh, L. (2004). A simple method for reconstructing a high-quality NDVI time-series data set based on the Savitzky-Golay filter. *Remote Sensing of Environment*, 91(3-4):332–344.

[Cheng and Wang, 2004] Cheng, J.-m. and Wang, H.-c. (2004). A method of estimating the equal error rate for automatic speaker verification. *SympoTIC '04. Joint 1st Workshop on Mobile Future & Symposium on Trends In Communications (IEEE Cat. No.04EX877)*, pages 285–288.

[Cheng et al., 2005] Cheng, O., Abdulla, W., and Salcic, Z. (2005). Performance Evaluation of Front-end Processing for Speech Recognition Systems. Technical Report 621, The University of Auckland.

[Church and de Oliveira, 2013] Church, K. and de Oliveira, R. (2013). What's up with whatsapp?: comparing mobile instant messaging behaviors with traditional SMS. *15th international conference on Human-computer interaction with mobile devices and services (MobileHCI'13)*, pages 352–361.

[Cisco, 2014] Cisco (2014). VNI Mobile Forecast Highlights.

[Cohen, 1988] Cohen, J. (1988). Statistical power analysis for the behavioral sciences.

[Cooley and Tukey, 1965] Cooley, J. W. and Tukey, J. W. (1965). An Algorithm for the Machine Computation of the Complex Fourier Series. *Mathematics of Computation*, 19:297.

[Coraggio, 1990] Coraggio, L. (1990). *Deleterious effects of intermittent interruptions on the task performance of knowledge workers: A laboratory investigation*. PhD thesis, University of Arizona.

[Cowie et al., 2001] Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., and Votsis, G. (2001). Emotion Recognition in Human-Computer Interaction. *Signal Processing Magazine*, 18(1):32–80.

[Cutrell et al., 2001] Cutrell, E., Czerwinski, M., and Horvitz, E. (2001). Notification, disruption, and memory: Effects of messaging interruptions on memory and performance. In *Proceedings of INTERACT '01*, number 1999, pages 263–269.

[Czerwinski et al., 2004] Czerwinski, M., Horvitz, E., and Wilhite, S. (2004). A diary study of task switching and interruptions. *Proceedings of the 2004 conference on Human factors in computing systems - CHI '04*, 6(1):175–182.

[Dauphin et al., 2014] Dauphin, Y. N., Chung, J., and Bengio BENGIOY, Y. (2014). RMSProp and equilibrated adaptive learning rates for non-convex optimization. pages 1–10.

[De Cooman, 2015] De Cooman, A. (2015). What Steps Data tells us about Country Lifestyles. In *http://blog.withings.com/2015/02/20/how-steps-data-tell-us-about-country-lifestyles/, accessed May 1, 2015*.

[Dean and Sridharan, 2010] Dean, D. and Sridharan, S. (2010). The QUT-NOISE-TIMIT Corpus for the Evaluation of Voice Activity Detection Algorithms. In *Proceedings of Interspeech 2010*, number September.

[Dehak, 2009] Dehak, N. (2009). *Discriminative and generative approaches for long- and short-term speaker characteristics modeling: application to speaker verification*. Phd thesis, Ecole de Technologie Superieure.

[Dehak et al., 2010] Dehak, N., Dehak, R., Glass, J., Reynolds, D., and Kenny, P. (2010). Cosine Similarity Scoring without Score Normalization Techniques. *Proceedings of Odyssey 2010 - The Speaker and Language Recognition Workshop (Odyssey 2010)*, pages 71–75.

[Dehak et al., 2009] Dehak, N., Kenny, P., Glembek, O., Dumouchel, P., and Burget, L. (2009). Support Vector Machines and Joint Factor Analysis for Speaker Verification. *Ieee Icassp*, (3):1–4.

[Dehak et al., 2011] Dehak, N., Kenny, P. J., Dehak, R., Dumouchel, P., and Ouellet, P. (2011). Front-end factor analysis for speaker verification. *IEEE Transactions on Audio, Speech and Language Processing*, 19(4):788–798.

[Dehak and Shum, 2011] Dehak, N. and Shum, S. (2011). Low-dimensional speech representation based on Factor Analysis and its applications. *Interspeech*.

[Dey, 2001] Dey, A. (2001). Understanding and using context. *Personal and ubiquitous computing*, 5(1):4–7.

[Díaz and Pedrero, 2006] Díaz, C. and Pedrero, A. (2006). Sound exposure during daily activities. *Applied acoustics*, 67(3):271–283.

[Dietterich, 1998] Dietterich, T. G. (1998). Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation*, 10(7):1895–1923.

[Dunn et al., 2001] Dunn, R. B., Quatieri, T. F., Reynolds, D. a., and Campbell, J. P. (2001). Speaker Recognition from Coded Speech in Matched and Mismatched Conditions. *ODYSSEY*, pages 1–6.

[Ekholm, 2015] Ekholm, J. (2015). Gartner Forecast: Mobile Data Traffic, Worldwide, 2011-2018. Technical report.

[Ekkekakis, 2012] Ekkekakis, P. (2012). Affect, mood, and emotion. *Measurement in sport and exercise psychology*, pages 321–332.

[Ekman, 2005] Ekman, P. (2005). Basic Emotions. In *Handbook of Cognition and Emotion*, volume 98, pages 45–60. John Wiley & Sons, Ltd, Chichester, UK.

[Esposito et al., 2008] Esposito, A., Avouris, N., and Hatzilygeroudis, I. (2008). *Verbal and Nonverbal Features of Human-Human and Human-Machine Interaction*. Springer Science & Business Media.

[Ester et al., 1996] Ester, M., Kriegel, H. P., Sander, J., and Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. *Second International Conference on Knowledge Discovery and Data Mining*, pages 226–231.

[Eyben and Weninger, 2013] Eyben, F. and Weninger, F. (2013). Real-life voice activity detection with LSTM Recurrent Neural Networks and an application to Hollywood movies. In *Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 483 – 487.

[Eyben et al., 2009] Eyben, F., Wöllmer, M., Graves, A., Schuller, B., Douglas-Cowie, E., and Cowie, R. (2009). On-line emotion recognition in a 3-D activation-valence-time continuum using acoustic and linguistic cues. *Journal on Multimodal User Interfaces*, 3(1-2):7–19.

[Eyben et al., 2012] Eyben, F., Wöllmer, M., and Schuller, B. (2012). A multitask approach to continuous five-dimensional affect sensing in natural speech. *ACM Transactions on Interactive Intelligent Systems*, 2(1):1–29.

[Falk et al., 2014] Falk, M., Hain, J., Marohn, F., Fischer, H., and Michel, R. (2014). *Statistik in Theorie und Praxis*. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Fan et al., 208] Fan, R.-e., Wang, X.-r., and Lin, C.-j. (208). LIBLINEAR: A Library for Large Linear Classification. *Journal of Machine Learning Research*, 9:1871–1874.

[Fawcett, 2006] Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874.

[Ferris et al., 2005] Ferris, D. P., Czerniecki, J. M., and Hannaford, B. (2005). An ankle-foot orthosis powered by artificial pneumatic muscles. *Journal of applied biomechanics*, 21(2):189–97.

[Finan et al., 1997] Finan, R. a., Sapeluk, a. T., and Damper, R. I. (1997). Impostor cohort selection for score normalisation in speaker verification. *Pattern Recognition Letters*, 18(9):881–888.

[Fischer et al., 2010] Fischer, J. E., Yee, N., Bellotti, V., Good, N., Benford, S., and Greenhalgh, C. (2010). Effects of content and time of delivery on receptivity to mobile interruptions. *MobileHCI '10: Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*, page 103.

[Fisher and Simmons, 2011] Fisher, R. and Simmons, R. (2011). Smartphone Interruptibility Using Density-Weighted Uncertainty Sampling with Reinforcement Learning. *2011 10th International Conference on Machine Learning and Applications and Workshops*, pages 436–441.

[Fletcher and McVeig, 1992] Fletcher, J. and McVeig, A. (1992). Towards A Model Of Segment And Syllable Duration In Australian English.

[Fogarty et al., 2005] Fogarty, J., Hudson, S. E., Atkeson, C. G., Avrahami, D., Forlizzi, J., Kiesler, S., Lee, J. C., and Yang, J. (2005). Predicting human interruptibility with sensors. *ACM Transactions on Computer-Human Interaction*, 12(1):119–146.

[Fogarty et al., 2004] Fogarty, J., Hudson, S. E., and Lai, J. (2004). Examining the robustness of sensor-based statistical models of human interruptibility. *Proceedings of the 2004 conference on Human factors in computing systems - CHI '04*, 6(1):207–214.

[Fontaine et al., 2007] Fontaine, J. R. J., Scherer, K. R., Roesch, E. B., and Ellsworth, P. C. (2007). The world of emotions is not two-dimensional. *Psychological Science*, 18(12):1050–1057.

[Furui, 2001] Furui, S. (2001). *Digital Speech Processing, Synthesis, and Recognition*. Marcel Dekker, New York, USA.

[Garcia-Romero and Espy-Wilson, 2011] Garcia-Romero, D. and Espy-Wilson, C. Y. (2011). Analysis of i-vector length normalization in speaker recognition systems. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 249–252.

[Garcia-Romero et al., 2012] Garcia-Romero, D., Zhou, X., and Espy-Wilson, C. Y. (2012). Multicondition training of Gaussian PLDA models in i-vector space for noise and reverberation robust speaker recognition. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 4257–4260.

[Garofolo et al., 1993] Garofolo, J., Lamel, L., Fisher, W., Fiscus, J., Pallett, D., Dahlgren, N., and Zue, V. (1993). TIMIT Acoustic-Phonetic Continuous Speech Corpus.

[Gillie and Broadbent, 1989] Gillie, T. and Broadbent, D. (1989). What makes interruptions disruptive? A study of length, similarity, and complexity. *Psychological Research*, 50(4):243–250.

[Glembek et al., 2009] Glembek, O., Burget, L., Dehak, N., Brümmer, N., and Kenny, P. (2009). Comparison of scoring methods used in speaker recognition with joint factor analysis. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 4057–4060.

[Gökçay and Yildirim, 2011] Gökçay, D. and Yildirim, G. (2011). *Affective Computing and Interaction*. IGI Global.

[Goldsmith and Baxter, 1996] Goldsmith, D. J. D. and Baxter, L. L. A. (1996). Constituting Relationships in Talk A Taxonomy of Speech Events in Social and Personal Relationships. *Human Communication Research*, 23(I):87–114.

[Gonina, 2013] Gonina, E. I. (2013). *A Framework for Productive, Efficient and Portable Parallel Computing*. PhD thesis, University of California, Berkeley.

[Gonzalez and Brookes, 2014] Gonzalez, S. and Brookes, M. (2014). A pitch estimation filter robust to high levels of noise (PEFAC). *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(2):518–530.

[Görür and Rasmussen, 2010] Görür, D. and Rasmussen, C. (2010). Dirichlet process Gaussian mixture models: Choice of the base distribution. *Journal of Computer Science and Technology*, 25(July):615–626.

[Grandjean et al., 2008] Grandjean, D., Sander, D., and Scherer, K. R. (2008). Conscious emotional experience emerges as a function of multilevel, appraisal-driven response synchronization. *Consciousness and Cognition*, 17(2):484–495.

[Graves, 2013] Graves, A. (2013). Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, pages 1–43.

[Greene and D'Oliveira, 1985] Greene, J. and D'Oliveira, M. (1985). *Learning to use statistical tests in psychology*.

[Groff et al., 1983] Groff, B. D., Baron, R. S., and Moore, D. L. (1983). Distraction, attentional conflict, and drivelike behavior. *Journal of Experimental Social Psychology*, 19(4):359–380.

[Grothendieck et al., 2009] Grothendieck, J., Gorin, A., and Borges, N. (2009). Social correlates of turn-taking behavior. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 4745–4748.

[Grundgeiger et al., 2010] Grundgeiger, T., Sanderson, P., MacDougall, H. G., and Venkatesh, B. (2010). Interruption management in the intensive care unit: Predicting resumption times and assessing distributed support. *Journal of experimental psychology. Applied*, 16(4):317–34.

[Gunes et al., 2011] Gunes, H., Schuller, B., Pantic, M., and Cowie, R. (2011). Emotion representation, analysis and synthesis in continuous space: A survey. *2011 IEEE International Conference on Automatic Face and Gesture Recognition and Workshops, FG 2011*, pages 827–834.

[Guo et al., 2002] Guo, Q., Wu, W., Massart, D., Boucon, C., and de Jong, S. (2002). Feature selection in principal component analysis of analytical data. *Chemometrics and Intelligent Laboratory Systems*, 61(1-2):123–132.

[Han et al., 2014] Han, K., Yu, D., and Tashev, I. (2014). Speech Emotion Recognition Using Deep Neural Network and Extreme Learning Machine. *Fifteenth Annual Conference of . . .* , (September):223–227.

[Hansson et al., 2001] Hansson, R., Ljungstrand, P., and Redström, J. (2001). Subtle and public notification cues for mobile devices. *Ubicomp 2001: Ubiquitous . . . .*

[Harris, 1978] Harris, F. (1978). On the use of windows for harmonic analysis with the discrete Fourier transform. *Proceedings of the IEEE*, 66(1):51–83.

[Hashimoto et al., 2013] Hashimoto, S., Tanaka, T., Aoki, K., and Fujita, K. (2013). Estimation of interruptibility during office work based on PC activity and conversation. *Human Interface and the Management of Information*, pages 297–306.

[Hatch et al., 2006] Hatch, A., Kajarekar, S., and Stolcke, A. (2006). Within-Class Covariance Normalization for SVM-based Speaker Recognition. In *Interspeech*, pages 2–5.

[Hemphill, 2003] Hemphill, J. F. (2003). Interpreting the magnitudes of correlation coefficients. *American Psychologist*, 58(1):78–79.

[Hermansky, 1990] Hermansky, H. (1990). Perceptual linear predictive (PLP) analysis of speech. *The Journal of the Acoustical Society of America*, 87(4):1738–1752.

[Hermansky and Morgan, 1994] Hermansky, H. and Morgan, N. (1994). RASTA processing of speech. *IEEE Transactions on Speech and Audio Processing*, 2(4):578–589.

[Herranz, 1999] Herranz, J. (1999). Model-Free Objective Bayesian Prediction. *Rev. Acad. Ciencias de Madrid*, 93(9):295–302.

[Ho and Intille, 2005] Ho, J. and Intille, S. S. (2005). Using context-aware computing to reduce the perceived burden of interruptions from mobile devices. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '05*, page 909.

[Hochreiter and Schmidhuber, 1997] Hochreiter, S. and Schmidhuber, J. (1997). Long short--term memory. *Neural Computation*, 9(8):1–32.

[Holm, 1979] Holm, S. (1979). A Simple Sequentially Rejective Multiple Test Procedure. *Scandinavian journal of statistics*, 6(2):65–70.

[Hönig et al., 2005] Hönig, F., Stemmer, G., Hacker, C., and Brugnara, F. (2005). Revising Perceptual Linear Prediction (PLP). In *INTERSPEECH 2005 - Eurospeech, 9th European Conference on Speech Communication and Technology*, pages 2997–3000.

[Horvitz and Apacible, 2003] Horvitz, E. and Apacible, J. (2003). Learning and reasoning about interruption. *Proceedings of the 5th international conference on Multimodal interfaces - ICMI '03*, page 20.

[Horvitz et al., 2005a] Horvitz, E., Apacible, J., and Subramani, M. (2005a). Balancing awareness and interruption: Investigation of notification deferral policies. In *UM'05 Proceedings of the 10th international conference on User Modeling*, pages 433–437.

[Horvitz et al., 1999] Horvitz, E., Jacobs, A., and Hovel, D. (1999). Attention-sensitive alerting. *UAI'99 Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, 98025:305–313.

[Horvitz et al., 2002] Horvitz, E., Koch, P., Kadie, C., and Jacobs, A. (2002). Coordinate: probabilistic forecasting of presence and availability. In *UAI'02 Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, number Im, pages 224–233.

[Horvitz et al., 2005b] Horvitz, E., Koch, P., and Sarin, R. (2005b). Bayesphone: Precomputation of Context-Sensitive Policies for Inquiry and Action in Mobile Devices. In *UM'05 Proceedings of the 10th international conference on User Modeling*, pages 251–260.

[Hsu et al., 2010] Hsu, C.-w., Chang, C.-c., and Lin, C.-j. (2010). A Practical Guide to Support Vector Classification. Technical Report 1, Department of Computer Science, National Taiwan University.

[Hudlicka and Gunes, 2012] Hudlicka, E. and Gunes, H. (2012). Benefits and limitations of continuous representations of emotions in affective computing: introduction to the special issue. *J. Synth. Emotions. v3 i1*.

[Ichikawa et al., 2005] Ichikawa, F., Chipchase, J., and Grignani, R. (2005). Where's the phone? A study of mobile phone location in public spaces. In *IEE Mobility Conference*

2005. The Second International Conference on Mobile Technology, Applications and Systems, volume 2005, pages 142–142. Iee.

[Iqbal and Bailey, 2008] Iqbal, S. T. and Bailey, B. P. (2008). Effects of intelligent notification management on users and their tasks. *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08*, page 93.

[Iqbal and Bailey, 2010] Iqbal, S. T. and Bailey, B. P. (2010). Oasis: A Framework for Linking Notication Delivery to the Perceptual Structure of Goal-Directed Tasks. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 17(4):1–28.

[Jiahong et al., 2006] Jiahong, Y., Liberman, M., and Cieri, C. (2006). Towards an integrated understanding of speaking rate in conversation. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2:541–544.

[Jiang et al., 2014] Jiang, Y., Lee, K., and Wang, L. (2014). PLDA in the i-supervector space for text-independent speaker verification. *EURASIP Journal on Audio, Speech, and Music Processing*, 2014:29.

[Juslin and Västfjäll, 2008] Juslin, P. N. and Västfjäll, D. (2008). Emotional responses to music: the need to consider underlying mechanisms. *The Behavioral and brain sciences*, 31(5):559–575; discussion 575–621.

[Kächele et al., 2014] Kächele, M., Schels, M., and Schwenker, F. (2014). Inferring Depression and Affect from Application Dependent Meta Knowledge. *Proceedings of the 4th ACM International Workshop on Audio/Visual Emotion Challenge (AVEC '14)*, pages 41–48.

[Kanagasundaram et al., 2011] Kanagasundaram, A., Vogt, R., Dean, D., Sridharan, S., and Mason, M. (2011). I-vector based speaker recognition on short utterances. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, (August):2341–2344.

[Kapoor and Horvitz, 2008] Kapoor, A. and Horvitz, E. (2008). Experience sampling for building predictive user models: a comparative study. In *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08*, page 657, New York, New York, USA. ACM Press.

[Kapoor et al., 2007] Kapoor, A., Horvitz, E., and Basu, S. (2007). Selective Supervision: Guiding Supervised Learning with Decision-Theoretic Active Learning. In *IJCAI'07 Proceedings of the 20th international joint conference on Artifical intelligence*, pages 877–882.

[Kendall, 2013] Kendall, T. (2013). *Speech Rate, Pause and Sociolinguistic Variation: Studies in Corpus Sociophonetics*. Palgrave Macmillan.

[Kenny et al., 2007] Kenny, P., Boulianne, G., Ouellet, P., and Dumouchel, P. (2007). Joint factor analysis versus eigenchannels in speaker recognition. *IEEE Transactions on Audio, Speech and Language Processing*, 15:1435–1447.

[Kenny et al., 2014] Kenny, P., Stafylakis, T., Ouellet, P., and Alam, M. J. (2014). JFA-based front ends for speaker recognition. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1705–1709.

[Kenny et al., 2013] Kenny, P., Stafylakis, T., Ouellet, P., Alam, M. J., and Dumouchel, P. (2013). PLDA for speaker verification with utterances of arbitrary duration. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, (1):7649–7653.

[Kern and Schiele, 2003] Kern, N. and Schiele, B. (2003). Context-aware notification for wearable computing. *Seventh IEEE International Symposium on Wearable Computers, 2003. Proceedings.*, pages 223–230.

[Kern and Schiele, 2006] Kern, N. and Schiele, B. (2006). Towards Personalized Mobile Interruptbility Estimation. In *Proc. 2nd Inter. Workshop on Location- and Context-Awareness (LoCa)*, pages 134–150.

[Khoury et al., 2014] Khoury, E., El Shafey, L., and Marcel, S. (2014). Spear: An open source toolbox for speaker recognition based on Bob. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2:1655–1659.

[Kinnunen, 2003] Kinnunen, T. (2003). *Spectral Features for Automatic Text-Independent Speaker Recognition*. Licentiate's thesis, University of Joensuu.

[Kinnunen et al., 2007] Kinnunen, T., Chernenko, E., Tuononen, M., Fränti, P., and Li, H. (2007). Voice Activity Detection Using MFCC Features and Support Vector Machine. *Int. Conf. on Speech and Computer (SPECOM07)*, 2(2):556–561.

[Kinnunen and Li, 2010] Kinnunen, T. and Li, H. (2010). An overview of text-independent speaker recognition: From features to supervectors. *Speech communication*, 52(1):12–40.

[Kinnunen and Rajan, 2013] Kinnunen, T. and Rajan, P. (2013). A practical, self-adaptive voice activity detector for speaker verification with noisy telephone and microphone data. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, volume 1, pages 7229–7233.

[Kreifeldt and McCarthy, 1981] Kreifeldt, J. G. and McCarthy, M. E. (1981). Interruption as a test of the user-computer interface. In *Proceedings of the 17th Annual Conference on Manual Control*, pages 655–667.

[Kruskal and Wish, 1978] Kruskal, J. B. and Wish, M. (1978). *Multidimensional Scaling*. SAGE Publications.

[Latorella, 1999] Latorella, K. (1999). Investigating interruptions: Implications for flightdeck performance. Technical Report October, NASA Langley Technical Report Server.

[Laver, 1994] Laver, J. (1994). *Principles of Phonetics*. Cambridge University Press.

[Lin and Weng, 2008] Lin, C.-j. and Weng, R. C. (2008). Trust Region Newton Method for Large-Scale Logistic Regression. *Journal of Machine Learning Research*, 9:627–650.

[Lopes, 2011] Lopes, R. H. C. (2011). Kolmogorov-Smirnov Test. In *International Encyclopedia of Statistical Science*, volume 30, pages 718–720. Springer Berlin Heidelberg, Berlin, Heidelberg.

[Lopez-Otero et al., 2014] Lopez-Otero, P., Docio-Fernandez, L., and Garcia-Mateo, C. (2014). iVectors for Continuous Emotion Recognition. *Iberspeech 2014: VIII Jornadas en Tecnologías del Habla and IV Iberian SLTech Workshop*.

[Lopez-tovar and Charalambous, 2015] Lopez-tovar, H. and Charalambous, A. (2015). Managing Smartphone Interruptions through Adaptive Modes and Modulation of Notifications. pages 296–299.

[Lu et al., 2011] Lu, H., Brush, A., Priyantha, B., Karlson, A., and Liu, J. (2011). SpeakerSense: energy efficient unobtrusive speaker identification on mobile phones. *Pervasive Computing*, pages 188–205.

[Lu et al., 2009] Lu, H., Pan, W., and Lane, N. (2009). SoundSense: scalable sound sensing for people-centric applications on mobile phones. In *MobiSys '09 Proceedings of the 7th international conference on Mobile systems, applications, and services*, pages 165–178.

[Madan and Pentland, 2006] Madan, A. and Pentland, A. (2006). VibeFones: Socially Aware Mobile Phones. *2006 10th IEEE International Symposium on Wearable Computers*, pages 109–112.

[Mark et al., 2008] Mark, G., Gudith, D., and Klocke, U. (2008). The cost of interrupted work. In *Proceeding of the twenty-sixth annual CHI conference on Human factors in computing systems - CHI '08*, page 107, New York, New York, USA. ACM Press.

[Marshall et al., 2009] Marshall, S. J., Levy, S. S., Tudor-Locke, C. E., Kolkhorst, F. W., Wooten, K. M., Ji, M., Macera, C. a., and Ainsworth, B. E. (2009). Translating Physical Activity Recommendations into a Pedometer-Based Step Goal. 3000 Steps in 30 Minutes. *American Journal of Preventive Medicine*, 36(5):410–415.

[Martin et al., 1997] Martin, A., Doddington, G., Kamm, T., Ordowski, M., and Przybocki, M. (1997). The DET Curve in Assessment of Detection Task Performance. *Proc. Eurospeech '97*, pages 1895–1898.

[Martin, 2004] Martin, D. W. (2004). *Doing Psychology Experiments*. Thomson/Wadsworth.

[Martínez and Kak, 2001] Martínez, A. and Kak, A. (2001). PCA versus LDA. *Pattern Analysis and Machine Intelligence*, 23(2):228–233.

[McCool et al., 2012] McCool, C., Marcel, S., Hadid, A., Pietikäinen, M., Matějka, P., Černocký, J., Poh, N., Kittler, J., Larcher, A., Lévy, C., Matrouf, D., Bonastre, J. F., Tresadern, P., and Cootes, T. (2012). Bi-modal person recognition on a mobile phone: Using mobile phone data. *Proceedings of the 2012 IEEE International Conference on Multimedia and Expo Workshops, ICMEW 2012*, pages 635–640.

[McFarlane and Latorella, 2002] McFarlane, D. and Latorella, K. (2002). The Scope and Importance of Human Interruption in Human-Computer Interaction Design. *Human-Computer Interaction*, 17(1):1–61.

[McKeown et al., 2012] McKeown, G., Valstar, M., Cowie, R., Pantic, M., and Schröder, M. (2012). The SEMAINE database: Annotated multimodal records of emotionally colored conversations between a person and a limited agent. *IEEE Transactions on Affective Computing*, 3(1):5–17.

[McLaren and Van Leeuwen, 2011] McLaren, M. and Van Leeuwen, D. (2011). Improved speaker recognition when using i-vectors from multiple speech sources. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 5460–5463.

[Mehl and Pennebaker, 2003] Mehl, M. R. and Pennebaker, J. W. (2003). The sounds of social life: A psychometric analysis of students' daily social environments and natural conversations. *Journal of Personality and Social Psychology*, 84(4):857–870.

[Mehl et al., 2001] Mehl, M. R., Pennebaker, J. W., Crow, D. M., Dabbs, J., and Price, J. H. (2001). The Electronically Activated Recorder (EAR): a device for sampling naturalistic daily activities and conversations. *Behavior research methods, instruments, & computers : a journal of the Psychonomic Society, Inc*, 33(4):517–523.

[Milner and Shao, 2002] Milner, B. and Shao, X. (2002). Speech reconstruction from mel-frequency cepstral coefficients using a source-filter model. . . . *Conference on Spoken Language Processing (ICSLP)*, pages 2421–2424.

[Moses and Randolph, 2004] Moses, P. S. and Randolph (2004). *Spectral Analysis of Signals*.

[Murphy, 2012] Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press, Cambridge, MA.

[Nagel et al., 2004] Nagel, K. S., Hudson, J. M., and Abowd, G. D. (2004). Predictors of availability in home life context-mediated communication. *Proceedings of the 2004 ACM conference on Computer supported cooperative work - CSCW '04*, 6(3):497.

[Nautsch, 2014] Nautsch, A. (2014). *Evaluation of text-independent speaker verification systems based on identity-vectors in short and variant duration scenarios*. PhD thesis, Hochschule Darmstadt.

[Nygren, 2009] Nygren, P. (2009). *Achieving equal loudness between audio files Evaluation and improvements of loudness algorithms*. PhD thesis, KTH Royal Institute of Technology.

[Oliver et al., 2004] Oliver, N., Garg, A., and Horvitz, E. (2004). Layered representations for learning and inferring office activity from multiple sensory channels. *Computer Vision and Image Understanding*, 96(2):163–180.

[Pan et al., 2003] Pan, Y.-X., Zhang, Z.-Z., Guo, Z.-M., Feng, G.-Y., Huang, Z.-D., and He, L. (2003). Application of pseudo amino acid composition for predicting protein subcellular location: stochastic signal processing approach. *Journal of protein chemistry*, 22(4):395–402.

[Panunzi et al., 2012] Panunzi, A., Raso, T., and Mello, H. (2012). *Pragmatics and Prosody: Illocution, Modality, Attitude, Information Patterning and Speech Annotation*. Firenze University Press.

[Parks and Burrus, 1987] Parks, T. W. and Burrus, C. S. (1987). Digital filter design.

[Pedregosa et al., 2011] Pedregosa, F., Weiss, R., and Brucher, M. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[Pentland, 2004] Pentland, A. (2004). Social Dynamics : Signals and Behavior. *Proceedings of the 3rd International Conference on Developmental Learning, Oct 2004*, 5:263–267.

[Pentland, 2008] Pentland, A. S. (2008). Honest Signals: How They Shape Our World.

[Pesaran and Shin, 1998] Pesaran, M. H. and Shin, Y. (1998). Generalized Impulse Response Analysis in Linear Multivariate Models. *Economics Letters*, 58(1):17–29.

[Pitt et al., 2007] Pitt, M., Dilley, L., Johnson, K., Kiesling, S., Raymond, W., Hume, E., and Fosler-Lussier, E. (2007). Buckeye Corpus of Conversational Speech (2nd release).

[Pitt et al., 2005] Pitt, M. a., Johnson, K., Hume, E., Kiesling, S., and Raymond, W. (2005). The Buckeye corpus of conversational speech: labeling conventions and a test of transcriber reliability. *Speech Communication*, 45(1):89–95.

[Plutchik, 1991] Plutchik, R. (1991). *The Emotions*.

[Prince and Elder, 2007] Prince, S. J. D. and Elder, J. H. (2007). Probabilistic linear discriminant analysis for inferences about identity. In *Proceedings of the IEEE International Conference on Computer Vision*, number iii.

[Rajan et al., 2013] Rajan, P., Kinnunen, T., and Hautamäki, V. (2013). Effect of multicondition training on i-vector PLDA configurations for speaker recognition. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, pages 3694–3697.

[Ramirez et al., 2007] Ramirez, J., Górriz, J., and Segura, J. (2007). Voice activity detection. fundamentals and speech recognition system robustness. In Grimm, M. and Kroschel, K., editors, *Robust Speech Recognition and Understanding*, number June, pages 1–23. I-TECH Education and Publishing.

[Ramirez et al., 2004] Ramirez, J., Segura, J., Benitez, C., Torre, a. D. L., and Rubio, a. (2004). Voice activity detection with noise reduction and long-term spectral divergence estimation. *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2(3):1093–1096.

[Refaeilzadeh et al., 2009] Refaeilzadeh, P., Tang, L., and Liu, H. (2009). Cross-Validation. In LIU, L. and ÖZSU, M. T., editors, *Encyclopedia of Database Systems*, pages 532–538. Springer US, Boston, MA.

[Reimers, 2010] Reimers, J. (2010). *Detection and Modeling of Social Situations via Distances and Shoulder-Angles*. PhD thesis, Technische Universität München.

[Reynolds, 2003] Reynolds, D. (2003). Channel robust speaker verification via feature mapping. *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, 2:53–56.

[Reynolds et al., 2000] Reynolds, D. a., Quatieri, T. F., and Dunn, R. B. (2000). Speaker Verification Using Adapted Gaussian Mixture Models. *Digital Signal Processing*, 10(1-3):19–41.

[Richert, 2013] Richert, W. (2013). *Building Machine Learning Systems with Python*. Packt Publishing, Birmingham.

[Richmond et al., 2008] Richmond, V. P., McCroskey, J. C., and Hickson, M. (2008). *Nonverbal Behavior in Interpersonal Relations*.

[Ristevski, 2015] Ristevski, Z. (2015). *Audio Contexts in Mobile Notification Management*. Bachelor's thesis, Technical University Munich.

[Robinson, 2001] Robinson, D. (2001). Replay Gain A Proposd Standard.

[Roda, 2011] Roda, C. (2011). Human attention and its implications for human computer interaction. In *Human Attention in Digital Environments*, pages 11–62. Cambridge University Press.

[Rosenthal et al., 2011] Rosenthal, S., Dey, A., and Veloso, M. (2011). Using Decision-Theoretic Experience Sampling to Build Personalized Mobile Phone Interruption Models. In *Pervasive'11 Proceedings of the 9th international conference on Pervasive computing*, pages 170–187.

[Rubinstein and Kroese, 2013] Rubinstein, R. Y. and Kroese, D. P. (2013). *The Cross-Entropy Method: A Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation and Machine Learning*. Springer Science & Business Media.

[Ruffin and King, 1999] Ruffin, C. and King, R. (1999). The analysis of hyperspectral data using Savitzky-Golay filtering. *IEEE 1999 International Geoscience and Remote Sensing Symposium. IGARSS'99 (Cat. No.99CH36293)*, 2(Part 1):1–3.

[Russell, 1980] Russell, J. A. (1980). A circumplex model of affect. *Journal of Personality and Social Psychology*, 39(6):1161–1178.

[Russell, 2009] Russell, J. A. (2009). Emotion, core affect, and psychological construction. *Cognition and emotion*, 23(7):12591283.

[Sacks et al., 1974] Sacks, H., Schegloff, E. a., and Jefferson, G. (1974). A simplest systematics for the organization of turn taking for conversation.

[Sadjadi et al., 2012] Sadjadi, S. O., Boil, H., and Hansen, J. H. L. (2012). A comparison of front-end compensation strategies for robust LVCSR under room reverberation and increased vocal effort. *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 4701–4704.

[Sakhnov et al., 2009] Sakhnov, K., Verteletskaya, E., and Simak, B. (2009). Approach for energy-based voice detector with adaptive scaling factor. *IAENG International Journal of Computer Science*, 36(November).

[Salamin et al., 2013] Salamin, H., Polychroniou, A., and Yinciarelli, A. (2013). Automatic recognition of personality and conflict handling style in mobile phone conversations. In *2013 14th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, pages 1–4. IEEE.

[Salzberg, 1997] Salzberg, S. (1997). On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data mining and knowledge discovery*, 328:317–328.

[Sarle, 2002] Sarle, W. S. (2002). comp.ai.neural-nets FAQ.

[Savitzky and Golay, 1964] Savitzky, A. and Golay, M. J. E. (1964). Smoothing and Differentiation of Data by Simplified Least Squares Procedures. *Analytical Chemistry*, 36(8):1627–1639.

[Sawhney and Schmandt, 2000] Sawhney, N. and Schmandt, C. (2000). Nomadic radio: speech and audio interaction for contextual messaging in nomadic environments. *ACM Transactions on Computer-Human Interaction*, 7(3):353–383.

[Schäfer, 2015] Schäfer, H. J. (2015). *Deriving Conversational Social Contexts from Audio-Data*. Master's thesis, Technical University Munich.

[Schisterman et al., 2005] Schisterman, E. F., Perkins, N. J., Liu, A., and Bondell, H. (2005). Optimal Cut-point and Its Corresponding Youden Index to Discriminate Individuals Using Pooled Blood Samples. *Epidemiology*, 16(1):73–81.

[Schuller et al., 2015] Schuller, B., Steidl, S., Batliner, A., Hantke, S., Florian, H., Orozco-arroyave, J. R., Elmar, N., Zhang, Y., and Weninger, F. (2015). The INTERSPEECH 2015 Computational Paralinguistics Challenge. *Interspeech*, pages 1–5.

[Schulze and Groh, 2014] Schulze, F. and Groh, G. (2014). Studying How Character of Conversation Affects Personal Receptivity to Mobile Notifications. In *Extended Abstracts of SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*, pages 1729–1734.

[Schulze and Groh, 2016] Schulze, F. and Groh, G. (2016). Conversational Context Helps Improve Mobile Notification Management. In *(UNDER REVIEW)*.

[Scikit-learn, 2012] Scikit-learn (2012). scikit-learn user guide 0.12.

[Seitle, 2014] Seitle, T. (2014). *Audio-Based Social Interaction Detection for Mobile Interruption Management*. Master's thesis, Technische Universität München.

[Setiono, 1995] Setiono, R. (1995). Chi2: feature selection and discretization of numeric attributes. In *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence*, pages 388–391. IEEE Comput. Soc. Press.

[Shen et al., 1998] Shen, J.-l., Hung, J.-w., and Lee, L.-s. (1998). Robust entropy-based endpoint detection for speech recognition in noisy environments. *ICSLP*, (1).

[Shouse, 2005] Shouse, E. (2005). Feeling, Emotion, Affect. *M/C Journal*, 8(6).

[Shum et al., 2010] Shum, S., Dehak, N., Dehak, R., and Glass, J. R. (2010). Unsupervised Speaker Adaptation based on the Cosine Similarity for Text-Independent Speaker Verification. *Proc. Odyssey*.

[Siewiorek et al., 2003] Siewiorek, D., Smailagic, A., Furukawa, J., Krause, A., Moraveji, N., Reiger, K., and Shaffer, J. (2003). SenSay: a context-aware mobile phone. *Proceedings of IEEE International Symposium on Wearable Computers*, pages 248–249.

[Sizov et al., 2014] Sizov, A., Lee, K. A., and Kinnunen, T. (2014). Unifying Probabilistic Linear Discriminant Analysis Variants in Biometric Authentication. In *Structural, Syntactic, and Statistical Pattern Recognition*, pages 464–475.

[Smith, 1997] Smith, S. W. (1997). The scientist and engineer's guide to digital signal processing.

[Smith-Lovin and Brody, 1989] Smith-Lovin, L. and Brody, C. (1989). Interruptions in Group Discussions: The Effects of Gender and Group Composition. *American Sociological Review*, 54(3):pp. 424–435.

[Smola and Scholkopf, 2004] Smola, a. J. and Scholkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, 14(3):199–222.

[Sohn, 1999] Sohn, J. (1999). A statistical model-based voice activity detection. *IEEE Signal Processing Letters*, 6(1):1–3.

[Speier et al., 2003] Speier, C., Vessey, I., and Valacich, J. S. (2003). The Effects of Interruptions, Task Complexity, and Information Presentation on Computer-Supported Decision-Making Performance. *Decision Sciences*, 34(4):771–797.

[Steidl, 2009] Steidl, S. (2009). *Automatic Classification of Emotion-Related User States in Spontaneous Children' s Speech*.

[Stevens et al., 1937] Stevens, S. S., Volkman, J., and Newman, E. (1937). A scale for the measurement of the psychological magnitude pitch. *The Journal of the Acoustical Society of America*, 8(3):185–190.

[Stivers et al., 2009] Stivers, T., Enfield, N. J., Brown, P., Englert, C., Hayashi, M., Heinemann, T., Hoymann, G., Rossano, F., de Ruiter, J. P., Yoon, K.-E., and Levinson, S. C. (2009). Universals and cultural variation in turn-taking in conversation. *Proceedings of the National Academy of Sciences of the United States of America*, 106(26):10587–10592.

[Strand and Egeberg, 2004] Strand, O. and Egeberg, A. (2004). Cepstral mean and variance normalization in the model domain. In *COST278 and ISCA Tutorial and Research Workshop (ITRW) on Robustness Issues in Conversational Interaction*, pages 2–5.

[Streefkerk et al., 2012] Streefkerk, J. W., McCrickard, D. S., van Esch-Bussemakers, M. P., and Neerincx, M. a. (2012). Balancing Awareness and Interruption in Mobile Patrol using Context-Aware Notification. *International Journal of Mobile Human Computer Interaction*, 4(3):1–27.

[SuperPowered, 2015] SuperPowered (2015). SuperPowered.

[Szeremeta and Zannin, 2009] Szeremeta, B. and Zannin, P. H. T. (2009). Analysis and evaluation of soundscapes in public parks through interviews and measurement of noise. *The Science of the total environment*, 407(24):6143–9.

[Ten Bosch et al., 2005] Ten Bosch, L., Oostdijk, N., and Boves, L. (2005). On temporal aspects of turn taking in conversational dialogues. *Speech Communication*, 47(1-2):80–86.

[The MathWorks Inc., 2014] The MathWorks Inc. (2014). MATLAB and Statistics Toolbox Release 2014a.

[Tian et al., 2015] Tian, L., Moore, J. D., and Lai, C. (2015). Emotion Recognition in Spontaneous and Acted Dialogues. *To Appear in the Proceedings of Affective Computing and Intelligent Interaction (ACII)*.

[Togneri and Pullella, 2011] Togneri, R. and Pullella, D. (2011). An overview of speaker identification: Accuracy and robustness issues. *Circuits and Systems Magazine, IEEE*, (May):23–61.

[Triggs, 2015] Triggs, R. (2015). A look at smartphone performance over the past 7 years.

[Tsuruoka et al., 2009] Tsuruoka, Y., Tsujii, J., and Ananiadou, S. (2009). Stochastic gradient descent training for L1-regularized log-linear models with cumulative penalty. *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1 - ACL-IJCNLP '09*, 1:477.

[Tudor-Locke et al., 2010] Tudor-Locke, C., Brashear, M. M., Johnson, W. D., and Katzmarzyk, P. T. (2010). Accelerometer profiles of physical activity and inactivity in normal weight, overweight, and obese U.S. men and women. *The international journal of behavioral nutrition and physical activity*, 7:60.

[Valstar et al., 2014] Valstar, M., Schuller, B., Smith, K., Almaev, T., Eyben, F., Krajewski, J., Cowie, R., and Pantic, M. (2014). AVEC 2014: 3D Dimensional Affect and Depression Recognition Challenge. *Proceedings of the 4th ACM International Workshop on Audio/Visual Emotion Challenge (AVEC '14)*, pages 3–10.

[van Dongen and Enright, 2012] van Dongen, S. and Enright, A. J. (2012). Metric distances derived from cosine similarity and Pearson and Spearman correlations. 2:2–6.

[Varma and Simon, 2006] Varma, S. and Simon, R. (2006). Bias in error estimation when using cross-validation for model selection. *BMC bioinformatics*, 7:91.

[Vinciarelli et al., 2009] Vinciarelli, A., Pantic, M., and Bourlard, H. (2009). Social signal processing: Survey of an emerging domain. *Image and Vision Computing*, 27(12):1743–1759.

[Wahib-ul haq, 2015] Wahib-ul haq, W. (2015). *Speaker Detection and Conversation Analysis on Mobile Devices*. Master's thesis, Technische Universität München.

[Weilhammer and Rabold, 2003] Weilhammer, K. and Rabold, S. (2003). Durational aspects in turn taking. *International Congresses of Phonetic Sciences*, (Vm Ii).

[Weninger and Bergmann, 2014] Weninger, F. and Bergmann, J. (2014). Introducing CURRENNT - the Munich Open-Source CUDA RecurREnt Neural Network Toolkit. *Journal of Machine Learning Research*, 99.

[Westphal, 1997] Westphal, M. (1997). The use of cepstral means in conversational speech recognition. In *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)*, pages 1143—-1146.

[Wieczorek, 2013] Wieczorek, M. (2013). *A Black-Box Approach to Interruption-Free Notifications*. Bachelor's thesis, Technical University Munich.

[Wöllmer, 2013] Wöllmer, M. (2013). *Context-Sensitive Machine Learning for Intelligent Human Behavior Analysis*. PhD thesis, Technical University Munich.

[Woo et al., 2006] Woo, R. H., Park, A., and Hazen, T. J. (2006). The MIT mobile device speaker verification corpus: Data collection and preliminary experiments. *IEEE Odyssey 2006: Workshop on Speaker and Language Recognition*, pages 1–6.

[Xu et al., 2013] Xu, C., Li, S., Liu, G., Zhang, Y., and Miluzzo, E. (2013). Crowd++ : Unsupervised Speaker Count with Smartphones. In *UbiComp '13 Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, pages 43–52.

[Yang and Kang, 2005] Yang, W. and Kang, J. (2005). Acoustic comfort evaluation in urban open public spaces. *Applied Acoustics*, 66(2):211–229.

[Zeng et al., 2009] Zeng, Z., Pantic, M., Roisman, G. I., and Huang, T. S. (2009). A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(1):39–58.

[Zwicker, 1961] Zwicker, E. (1961). Subdivision of the Audible Frequency Range into Critical Bands. *The Journal Of The Acoustical Society Of America*, 33(2):248.

A. Taxonomy Questionnaire

"The goal of this questionnaire is to determine the level of reachability (via any kind of a mobile communication device, eg. smartphone) of individuals during different speech events, which they might take part in.
Reachability means: how acceptable it is for you, with your environment in mind, to be reached at that moment.
The scale ranges from 1 (least reachable) to 5 (completely reachable), where a good guideline would be: 1 - no notification, 2 - LED notification, 3 - vibrations, 4 - silent ringtone/beep, 5 - normal ringtone/beep
Furthermore, in order to assess each answer's relevance, you will have to assign a weight to each answer, depending on how often you find yourself in the respective situation.
The second scale ranges from 1 (never) to 6 (very frequently):
1 - never, 2 - very rarely, 3 - rarely, 4 - occasionally, 5 - frequently, 6 - very frequently
The questionnaire is anonymous, which means the collected data will only be used to create a statistic in order to classify the situations mentioned below."

Speech event	Personal reachability						Frequency of occurrence					
Superficial talk:												
- small talk (a kind of talk to pass time and avoid being rude)	1	2	3	4	5	n/a	1	2	3	4	5	6
- sports talk (the kind of talk that occurs while playing or watching a sporting event)	1	2	3	4	5	n/a	1	2	3	4	5	6
- current events talk (a conversation in which the topic is limited to talking about news and current events)	1	2	3	4	5	n/a	1	2	3	4	5	6
Informal talk:												
- gossip (exchanging opinions or information about someone else when that person isn't present)	1	2	3	4	5	n/a	1	2	3	4	5	6
- joking around (a playful kind of talk to have fun or release tension)	1	2	3	4	5	n/a	1	2	3	4	5	6
- catching up (the kind of conversation you have when you haven't talked with someone recently, and you talk about the events in your lives that have occurred since you last spoke)	1	2	3	4	5	n/a	1	2	3	4	5	6
- recapping the day's events (telling about what's up and what happened to each person during the day)	1	2	3	4	5	n/a	1	2	3	4	5	6
- getting to know someone (the kind of small talk you have when you want to be friendly and get acquainted with someone)	1	2	3	4	5	n/a	1	2	3	4	5	6
- morning talk (the kind of routine talk you have when you first wake up in the morning)	1	2	3	4	5	n/a	1	2	3	4	5	6
- bedtime talk (the kind of routine talk you have right before you go to bed)	1	2	3	4	5	n/a	1	2	3	4	5	6
Involving and positive talk:												

- reminiscing (talking with someone about shared events you experienced together in the past)	1	2	3	4	5	n/a	1	2	3	4	5	6
- making up (conversations in which one or both people apologize for violating some expectations)	1	2	3	4	5	n/a	1	2	3	4	5	6
- love talk (talk that has little content but expresses love and gives attention and affection)	1	2	3	4	5	n/a	1	2	3	4	5	6
- positive relationship talk (talking about the nature and state of a relationship)	1	2	3	4	5	n/a	1	2	3	4	5	6
Negatively valenced talk:												
- conflict (conversations in which the two people disagree)	1	2	3	4	5	n/a	1	2	3	4	5	6
- serious conversation (a two-way, in-depth discussion or exchange of feelings, opinions, or ideas about some personal and important topic)	1	2	3	4	5	n/a	1	2	3	4	5	6
- talking about problems (a conversation in which one person tells about some problem he or she is having and the other person tries to help)	1	2	3	4	5	n/a	1	2	3	4	5	6
- breaking bad news (a conversation in which one person doesn't know about something bad that has happened and another person tells them the bad news)	1	2	3	4	5	n/a	1	2	3	4	5	6
- complaining (expressing negative feelings, frustrations, gripes, or complaints about some common experience where negative feelings are directed toward the topic but not toward the other people in the conversation)	1	2	3	4	5	n/a	1	2	3	4	5	6
- negative relationship talk (talking about the nature and state of a relationship)	1	2	3	4	5	n/a	1	2	3	4	5	6
Formal and goal-directed talk:												
- group discussion (talk to exchange information, persuade other people, or make decisions when a group is gathered)	1	2	3	4	5	n/a	1	2	3	4	5	6
- persuading conversation (conversation in which one person has the goal of convincing the other person to do something)	1	2	3	4	5	n/a	1	2	3	4	5	6
- decision-making conversation (conversation in which people have the goal of making a decision about some task)	1	2	3	4	5	n/a	1	2	3	4	5	6
- giving/getting instructions (a conversation in which one person gives another person information or directions about how to do some task)	1	2	3	4	5	n/a	1	2	3	4	5	6
- class information talk (informal conversations in which you find out about class assignments, exams, or course material)	1	2	3	4	5	n/a	1	2	3	4	5	6

- lecture (a one-way kind of conversation in which one person tells another person how to act or what to do)	1	2	3	4	5	n/a	1	2	3	4	5	6
- interrogation (a one-way kind of conversation in which one person grills the other person with questions)	1	2	3	4	5	n/a	1	2	3	4	5	6
Less formal goal-directed talk:												
- making plans (talking to arrange a meeting or arrange to do something with someone)	1	2	3	4	5	n/a	1	2	3	4	5	6
- asking a favor (talk with the specific purpose of getting someone to do something for you)	1	2	3	4	5	n/a	1	2	3	4	5	6
- asking out (the kind of talk when one person asks another person out on a date)	1	2	3	4	5	n/a	1	2	3	4	5	6

B. Taxonomy Evaluation

```
>>> stats.friedmanchisquare(experimentDF['Superficial talk'], experimentDF['Informal talk'], experimentDF['Involving and
      positive talk'], experimentDF['Negatively valenced talk'], experimentDF['Formal and goal-directed talk'],
      experimentDF['Less formal goal-directed talk'])
>>> p_list = []
>>> for i in range(len(experimentDF.columns.values)):
...     for j in range(i+1,len(experimentDF.columns.values)):
...             print 'Now comparing situation \'%s\' with situation \'%s\'' % (experimentDF.columns.values[i],
      experimentDF.columns.values[j])
...             stats.wilcoxon(experimentDF[experimentDF.columns.values[i]], experimentDF[experimentDF.columns.values[j]])
...
(175.3468354430378, 5.2722146210378925e-36)
Now comparing situation 'Superficial talk' with situation 'Informal talk'
(216.5, 3.5135613263323865e-06)
Now comparing situation 'Superficial talk' with situation 'Involving and positive talk'
(12.0, 1.4306931845803308e-10)
Now comparing situation 'Superficial talk' with situation 'Negatively valenced talk'
(16.5, 1.2206993696371506e-10)
Now comparing situation 'Superficial talk' with situation 'Formal and goal-directed talk'
(2.0, 5.7062169600658888e-11)
Now comparing situation 'Superficial talk' with situation 'Less formal goal-directed talk'
(58.5, 9.1367037137132691e-09)
Now comparing situation 'Informal talk' with situation 'Involving and positive talk'
(45.5, 5.4361724437713947e-10)
Now comparing situation 'Informal talk' with situation 'Negatively valenced talk'
(13.5, 1.0490448191638121e-10)
Now comparing situation 'Informal talk' with situation 'Formal and goal-directed talk'
(10.0, 8.7187628936168313e-11)
Now comparing situation 'Informal talk' with situation 'Less formal goal-directed talk'
(240.0, 8.9355093745796235e-06)
Now comparing situation 'Involving and positive talk' with situation 'Negatively valenced talk'
(223.0, 2.7033804823437234e-06)
Now comparing situation 'Involving and positive talk' with situation 'Formal and goal-directed talk'
(209.0, 9.2572037584112372e-07)
Now comparing situation 'Involving and positive talk' with situation 'Less formal goal-directed talk'
(511.0, 0.029920610743119667)
Now comparing situation 'Negatively valenced talk' with situation 'Formal and goal-directed talk'
(706.0, 0.45288680999021413)
Now comparing situation 'Negatively valenced talk' with situation 'Less formal goal-directed talk'
(165.0, 6.5549199598712932e-07)
Now comparing situation 'Formal and goal-directed talk' with situation 'Less formal goal-directed talk'
(194.5, 8.4806805710186242e-07)

Sorted according to p-value; corresponding Holm-corrected alpha; p<alpha ?

5.7062169600658888e-11, 0.0033333333333333335, YES
8.7187628936168313e-11, 0.0035714285714285718, YES
1.0490448191638121e-10, 0.0038461538461538464, YES
1.2206993696371506e-10, 0.004166666666666667, YES
1.4306931845803308e-10, 0.004545454545454546, YES
5.4361724437713947e-10, 0.005, YES
9.1367037137132691e-09, 0.005555555555555556, YES
6.5549199598712932e-07, 0.00625, YES
8.4806805710186242e-07, 0.0071428571428571435, YES
9.2572037584112372e-07, 0.008333333333333333, YES
2.7033804823437234e-06, 0.01, YES
3.5135613263323865e-06, 0.0125, YES
8.9355093745796235e-06, 0.016666666666666666, YES
0.029920610743119667, 0.025, NO ['Involving and positive talk' vs 'Less formal goal-directed talk']
0.45288680999021413, 0.05, NO ['Negatively valenced talk' vs 'Formal and goal-directed talk']
```

Listing B.1: *Output of statistical tests*

C. Voice Activity Detection Performance

```
>>> execfile('run_vad_proba_analysis_script.py')
Checking whether the samples were NOT drawn from a normal distribution (H0)

SVM: p is 0.640.
RF: p is 0.251.
GMM: p is 0.298.
NN: p is 0.286.
GNB: p is 0.742.

SVM vs RF: The t-statistic is 37.632 and the p-value is 0.00000.
SVM vs GMM: The t-statistic is 86.836 and the p-value is 0.00000.
SVM vs LSTM: The t-statistic is 12.788 and the p-value is 0.00000.
SVM vs GNB: The t-statistic is 132.371 and the p-value is 0.00000.
RF vs GMM: The t-statistic is 17.231 and the p-value is 0.00000.
RF vs LSTM: The t-statistic is -1.535 and the p-value is 0.15913.
RF vs GNB: The t-statistic is 87.846 and the p-value is 0.00000.
GMM vs LSTM: The t-statistic is -8.758 and the p-value is 0.00001.
GMM vs GNB: The t-statistic is 47.718 and the p-value is 0.00000.
NN vs GNB: The t-statistic is 27.559 and the p-value is 0.00000.
```

Listing C.1: *Output of statistical tests*

D. Speaker Detection Performance

D.1. Experiment E1

D.1.1. GMM

```
  EER:
   Overall mean EER is 0.32 with std 0.04
   Worst speaker is 5 with EER 0.37
   Best speaker is 0 with EER 0.24
   Overall mean IR-specific EER is 0.15 (not global!)
   Standard deviation of EER between IRs is 0.07 (not global!)
   Worst dataset is 0 with 0.26 (not global!)
   Best dataset is 1 with 0.08 (not global!)
  Thresholds:
   Mean IR-specific threshold for IR #0 is 7.26 with std 1.17 (not global!)
   Mean IR-specific threshold for IR #1 is 6.56 with std 1.03 (not global!)
   Mean IR-specific threshold for IR #2 is 10.72 with std 0.86 (not global!)
   Mean IR-specific threshold for IR #3 is 5.94 with std 0.96 (not global!)
   Mean (over speakers) of standard deviation (over IRs) of threshold is 1.91 (not global!)
   Standard deviation of final thresholds is 0.82

>>> global_confusion_matrix
[[173042, 4905], [91067, 8986]]
```

D.1.2. TV cosine

```
  EER:
   Overall mean EER is 0.36 with std 0.07
   Worst speaker is 5 with EER 0.47
   Best speaker is 3 with EER 0.24
   Overall mean IR-specific EER is 0.21 (not global!)
   Standard deviation of EER between IRs is 0.01 (not global!)
   Worst dataset is 0 with 0.22 (not global!)
   Best dataset is 1 with 0.19 (not global!)
  Thresholds:
   Mean IR-specific threshold for IR #0 is 1.24 with std 0.27 (not global!)
   Mean IR-specific threshold for IR #1 is 2.03 with std 0.55 (not global!)
   Mean IR-specific threshold for IR #2 is 2.78 with std 0.72 (not global!)
   Mean IR-specific threshold for IR #3 is 1.61 with std 0.38 (not global!)
   Mean (over speakers) of standard deviation (over IRs) of threshold is 0.58 (not global!)
   Standard deviation of final thresholds is 0.34

>>> global_confusion_matrix
[[161730, 5695], [102419, 8156]]
```

D.1.3. TV PLDA

```
  EER:
   Overall mean EER is 0.25 with std 0.07
   Worst speaker is 6 with EER 0.37
   Best speaker is 1 with EER 0.09
   Overall mean IR-specific EER is 0.09 (not global!)
   Standard deviation of EER between IRs is 0.05 (not global!)
```

```
    Worst dataset is 0 with 0.17 (not global!)
    Best dataset is 2 with 0.05 (not global!)
   Thresholds:
    Mean IR-specific threshold for IR #0 is 4.25 with std 0.38 (not global!)
    Mean IR-specific threshold for IR #1 is 4.04 with std 0.50 (not global!)
    Mean IR-specific threshold for IR #2 is 5.69 with std 0.38 (not global!)
    Mean IR-specific threshold for IR #3 is 3.90 with std 0.43 (not global!)
    Mean (over speakers) of standard deviation (over IRs) of threshold is 0.75 (not global!)
    Standard deviation of final thresholds is 0.31

>>> global_confusion_matrix
[[195000, 4038], [69088, 9874]]
```

E. Evaluation

E.1. Comparison of Affective and Social Properties

Listing E.1: *Results of a linear regression with cut-off value 3s*

```
>>> scipy.stats.pearsonr(formlist, depthlist)
(0.44903972592515906, 1.059538502970626e-156)
>>>
>>> scipy.stats.pearsonr(formlist, actlist)
(-0.079742874896584978, 7.1645162777420164e-06)
>>> scipy.stats.pearsonr(formlist, powerlist)
(0.1256658069448448, 1.335223311427177e-12)
>>> scipy.stats.pearsonr(formlist, novlist)
(0.16249735596635015, 3.798946871919697e-20)
>>> scipy.stats.pearsonr(formlist, vallist)
(-0.18474386126367329, 1.1497343728965807e-25)
>>>
>>> scipy.stats.pearsonr(depthlist, actlist)
(-0.12551314181939466, 1.4216357590579875e-12)
>>> scipy.stats.pearsonr(depthlist, powerlist)
(0.097726649954149369, 3.6818030745080808e-08)
>>> scipy.stats.pearsonr(depthlist, novlist)
(-0.0020331759989521193, 0.90902717555658263)
>>> scipy.stats.pearsonr(depthlist, vallist)
(-0.19006825062153879, 4.2626733873242802e-27)
>>>
>>> scipy.stats.pearsonr(powerlist, novlist)
(0.26810493949305653, 3.5694735904484869e-53)
>>> scipy.stats.pearsonr(actlist, novlist)
(-0.17008683876016112, 6.0189201954130593e-22)
>>> scipy.stats.pearsonr(vallist, novlist)
(-0.29618581502113223, 4.9777041708374573e-65)
>>>
>>> scipy.stats.pearsonr(powerlist, actlist)
(-0.28171900539076011, 9.4725496555858899e-59)
>>> scipy.stats.pearsonr(powerlist, vallist)
(-0.24796560120712491, 1.6793983236672264e-45)
>>>
>>> scipy.stats.pearsonr(actlist, vallist)
(0.29258829237658113, 1.9639972535583739e-63)
>>>
>>> scipy.stats.pearsonr(vallist, vallist_type)
(-0.039031969900711366, 0.028202804618900131)
>>>
>>> data = {}
>>> data['formality'] = formlist
>>> data['depth'] = depthlist
>>> data['power'] = powerlist
>>> data['novelty'] = novlist
>>> data['activity'] = actlist
>>> data['valence'] = vallist
>>> data['val_social'] = vallist_type
>>>
>>> from pandas import DataFrame
>>> df = DataFrame(data)
>>> df.corr()
activity     depth  formality   novelty     power  val_social  \
activity    1.000000 -0.125513  -0.079743 -0.170087 -0.281719   -0.008992
depth      -0.125513  1.000000   0.449040 -0.002033  0.097727    0.129190
formality  -0.079743  0.449040   1.000000  0.162497  0.125666    0.514977
novelty    -0.170087 -0.002033   0.162497  1.000000  0.268105    0.066917
```

```
power       −0.281719  0.097727   0.125666  0.268105  1.000000    0.077833
val_social  −0.008992  0.129190   0.514977  0.066917  0.077833    1.000000
valence      0.292588 −0.190068  −0.184744 −0.296186 −0.247966   −0.039032

valence
activity     0.292588
depth       −0.190068
formality   −0.184744
novelty     −0.296186
power       −0.247966
val_social  −0.039032
valence      1.000000
>>>
>>> import statsmodels.api as sm
>>> from statsmodels.formula.api import ols
>>>
>>> model = ols("formality ~ * expectation * activity * valence", data=data).fit()
>>> print model.summary()
OLS Regression Results
==============================================================================
Dep. Variable:              formality   R-squared:                       0.059
Model:                            OLS   Adj. R-squared:                  0.055
Method:                 Least Squares   F-statistic:                     13.21
Date:                Tue, 06 Oct 2015   Prob (F-statistic):           6.83e-33
Time:                        18:02:18   Log-Likelihood:                -13092.
No. Observations:                3161   AIC:                          2.622e+04
Df Residuals:                    3145   BIC:                          2.631e+04
Df Model:                          15
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------------------------
Intercept                       54.2461      5.481      9.898      0.000        43.500    64.992
power                           -0.0199      0.102     -0.195      0.845        -0.219     0.179
novelty                         -0.1633      0.085     -1.914      0.056        -0.331     0.004
power:novelty                    0.0011      0.001      0.787      0.431        -0.002     0.004
activity                        -0.2839      0.083     -3.405      0.001        -0.447    -0.120
power:activity                   0.0017      0.002      1.024      0.306        -0.002     0.005
novelty:activity                 0.0044      0.001      3.449      0.001         0.002     0.007
power:novelty:activity       -2.619e-05   2.34e-05     -1.118      0.264     -7.21e-05  1.98e-05
valence                         -0.2670      0.076     -3.530      0.000        -0.415    -0.119
power:valence                    0.0006      0.001      0.420      0.674        -0.002     0.003
novelty:valence                  0.0031      0.001      2.287      0.022         0.000     0.006
power:novelty:valence        -1.067e-05   2.45e-05     -0.435      0.663     -5.87e-05  3.74e-05
activity:valence                 0.0039      0.001      3.649      0.000         0.002     0.006
power:activity:valence       -2.382e-05   2.16e-05     -1.101      0.271     -6.63e-05  1.86e-05
novelty:activity:valence     -6.328e-05   1.87e-05     -3.377      0.001        -0.000 -2.65e-05
power:novelty:activity:valence  3.67e-07   3.53e-07      1.039      0.299     -3.26e-07  1.06e-06
==============================================================================
Omnibus:                      938.450   Durbin-Watson:                   1.495
Prob(Omnibus):                  0.000   Jarque-Bera (JB):             2198.162
Skew:                           1.676   Prob(JB):                         0.00
Kurtosis:                       5.335   Cond. No.                     1.86e+08
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 1.86e+08. This might indicate that there are
strong multicollinearity or other numerical problems.
>>>
>>> model = ols("depth ~ * expectation * activity * valence", data=data).fit()
>>> print model.summary()
OLS Regression Results
==============================================================================
Dep. Variable:                  depth   R-squared:                       0.058
Model:                            OLS   Adj. R-squared:                  0.053
Method:                 Least Squares   F-statistic:                     12.85
Date:                Tue, 06 Oct 2015   Prob (F-statistic):           7.25e-32
Time:                        18:02:18   Log-Likelihood:                -13973.
No. Observations:                3161   AIC:                          2.798e+04
Df Residuals:                    3145   BIC:                          2.807e+04
Df Model:                          15
Covariance Type:            nonrobust
================================================================================================
                                   coef    std err          t      P>|t|      [95.0% Conf. Int.]
```

```
                                coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept                     22.6164      7.242      3.123      0.002       8.418      36.815
power                          0.5134      0.134      3.823      0.000       0.250       0.777
novelty                        0.1359      0.113      1.206      0.228      −0.085       0.357
power:novelty                 −0.0047      0.002     −2.450      0.014      −0.008      −0.001
activity                       0.1047      0.110      0.951      0.342      −0.111       0.321
power:activity                −0.0048      0.002     −2.154      0.031      −0.009      −0.000
novelty:activity              −0.0019      0.002     −1.133      0.257      −0.005       0.001
power:novelty:activity      4.952e−05    3.1e−05      1.599      0.110   −1.12e−05       0.000
valence                        0.1774      0.100      1.776      0.076      −0.019       0.373
power:valence                 −0.0085      0.002     −4.436      0.000      −0.012      −0.005
novelty:valence               −0.0028      0.002     −1.601      0.109      −0.006       0.001
power:novelty:valence       7.76e−05    3.24e−05      2.396      0.017    1.41e−05       0.000
activity:valence              −0.0031      0.001     −2.191      0.028      −0.006      −0.000
power:activity:valence      9.485e−05    2.86e−05      3.317      0.001    3.88e−05       0.000
novelty:activity:valence    2.996e−05    2.48e−05      1.210      0.226   −1.86e−05    7.85e−05
power:novelty:activity:valence −9.282e−07    4.67e−07     −1.989      0.047   −1.84e−06   −1.31e−08
==============================================================================
Omnibus:                      179.349   Durbin−Watson:                   1.618
Prob(Omnibus):                  0.000   Jarque−Bera (JB):              208.532
Skew:                           0.623   Prob(JB):                    5.22e−46
Kurtosis:                       2.829   Cond. No.                    1.86e+08
==============================================================================
```

Listing E.2: *Results of a linear regression with cut-off value 12s*

```
>>> scipy.stats.pearsonr(formlist, depthlist)
(0.42873141287849226, 3.3657957646902214e−33)
>>>
>>> scipy.stats.pearsonr(formlist, actlist)
(−0.06279601345799822, 0.094068664300586982)
>>> scipy.stats.pearsonr(formlist, powerlist)
(0.14130581222237854, 0.00015502333264610512)
>>> scipy.stats.pearsonr(formlist, novlist)
(0.25790149948414132, 2.7857053781020626e−12)
>>> scipy.stats.pearsonr(formlist, vallist)
(−0.11766932332800091, 0.0016595900672746547)
>>>
>>> scipy.stats.pearsonr(depthlist, actlist)
(−0.088816122777072015, 0.017767323969451784)
>>> scipy.stats.pearsonr(depthlist, powerlist)
(0.17891238514944055, 1.552112593718851e−06)
>>> scipy.stats.pearsonr(depthlist, novlist)
(0.027719853914942361, 0.4602105072658178)
>>> scipy.stats.pearsonr(depthlist, vallist)
(−0.17707130605232144, 1.9926637629376016e−06)
>>>
>>> scipy.stats.pearsonr(powerlist, novlist)
(0.21720903988607485, 4.7492514401912062e−09)
>>> scipy.stats.pearsonr(actlist, novlist)
(−0.139272788945175, 0.00019307054219834966)
>>> scipy.stats.pearsonr(vallist, novlist)
(−0.31584766852968643, 5.8692318086013981e−18)
>>>
>>> scipy.stats.pearsonr(powerlist, actlist)
(−0.34910349484104897, 7.7933288425051462e−22)
>>> scipy.stats.pearsonr(powerlist, vallist)
(−0.21575014328500658, 6.0474002593085584e−09)
>>>
>>> scipy.stats.pearsonr(actlist, vallist)
(0.27322070283080341, 1.1778805349701851e−13)
>>>
>>> scipy.stats.pearsonr(vallist, vallist_type)
(0.06451983611020122, 0.085364876269014192)
>>>
>>> data = {}
>>> data['formality'] = formlist
>>> data['depth'] = depthlist
>>> data['power'] = powerlist
>>> data['novelty'] = novlist
>>> data['activity'] = actlist
>>> data['valence'] = vallist
>>> data['val_social'] = vallist_type
>>>
```

```
>>> from pandas import DataFrame
>>> df = DataFrame(data)
>>> df.corr()
activity      depth  formality   novelty     power  val_social  \
activity     1.000000 -0.088816 -0.062796 -0.139273 -0.349103   -0.029167
depth       -0.088816  1.000000  0.428731  0.027720  0.178912    0.147709
formality   -0.062796  0.428731  1.000000  0.257901  0.141306    0.568043
novelty     -0.139273  0.027720  0.257901  1.000000  0.217209    0.173183
power       -0.349103  0.178912  0.141306  0.217209  1.000000    0.084712
val_social  -0.029167  0.147709  0.568043  0.173183  0.084712    1.000000
valence      0.273221 -0.177071 -0.117669 -0.315848 -0.215750    0.064520

valence
activity     0.273221
depth       -0.177071
formality   -0.117669
novelty     -0.315848
power       -0.215750
val_social   0.064520
valence      1.000000
>>>
>>> import statsmodels.api as sm
>>> from statsmodels.formula.api import ols
>>>
>>> model = ols("formality ~ * expectation * activity * valence", data=data).fit()
>>> print model.summary()
OLS Regression Results
==============================================================================
Dep. Variable:              formality   R-squared:                       0.167
Model:                            OLS   Adj. R-squared:                  0.149
Method:                 Least Squares   F-statistic:                     9.284
Date:                Tue, 06 Oct 2015   Prob (F-statistic):           4.99e-20
Time:                        18:01:48   Log-Likelihood:                -2980.4
No. Observations:                 712   AIC:                             5993.
Df Residuals:                     696   BIC:                             6066.
Df Model:                          15
Covariance Type:            nonrobust
================================================================================================
coef    std err          t      P>|t|      [95.0% Conf. Int.]
------------------------------------------------------------------------------------------------
Intercept                       28.1934     15.861      1.778      0.076        -2.948    59.335
power                            0.5347      0.259      2.063      0.039         0.026     1.043
novelty                          0.0547      0.247      0.222      0.825        -0.430     0.539
power:novelty                   -0.0056      0.004     -1.465      0.143        -0.013     0.002
activity                        -0.3327      0.241     -1.380      0.168        -0.806     0.141
power:activity                  -0.0032      0.004     -0.763      0.446        -0.012     0.005
novelty:activity                 0.0100      0.004      2.665      0.008         0.003     0.017
power:novelty:activity        7.961e-06   6.68e-05      0.119      0.905        -0.000     0.000
valence                          0.2939      0.235      1.251      0.211        -0.167     0.755
power:valence                   -0.0093      0.004     -2.368      0.018        -0.017    -0.002
novelty:valence                 -0.0055      0.004     -1.305      0.192        -0.014     0.003
power:novelty:valence            0.0002   6.73e-05      2.660      0.008      4.68e-05     0.000
activity:valence                 0.0022      0.003      0.671      0.502        -0.004     0.008
power:activity:valence           5.93e-05   5.52e-05      1.074      0.283     -4.91e-05     0.000
novelty:activity:valence       -7.665e-05   5.92e-05     -1.295      0.196        -0.000  3.96e-05
power:novelty:activity:valence -9.795e-07   1.01e-06     -0.967      0.334     -2.97e-06  1.01e-06
==============================================================================
Omnibus:                      134.188   Durbin-Watson:                   1.260
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              216.752
Skew:                           1.210   Prob(JB):                     8.57e-48
Kurtosis:                       4.202   Cond. No.                     2.14e+08
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.14e+08. This might indicate that there are
strong multicollinearity or other numerical problems.
>>>
>>> model = ols("depth ~ * expectation * activity * valence", data=data).fit()
>>> print model.summary()
OLS Regression Results
==============================================================================
Dep. Variable:                  depth   R-squared:                       0.112
Model:                            OLS   Adj. R-squared:                  0.092
Method:                 Least Squares   F-statistic:                     5.823
```

```
Date:                Tue, 06 Oct 2015   Prob (F-statistic):          1.93e-11
Time:                        18:01:48   Log-Likelihood:                -3149.8
No. Observations:                 712   AIC:                             6332.
Df Residuals:                     696   BIC:                             6405.
Df Model:                          15
Covariance Type:            nonrobust
================================================================================
coef    std err          t      P>|t|      [95.0% Conf. Int.]
--------------------------------------------------------------------------------
Intercept                    -12.3073     20.120     -0.612      0.541     -51.811     27.197
power                          1.1871      0.329      3.611      0.000       0.542      1.832
novelty                        0.6315      0.313      2.017      0.044       0.017      1.246
power:novelty                 -0.0129      0.005     -2.673      0.008      -0.022     -0.003
activity                       0.3859      0.306      1.262      0.208      -0.215      0.987
power:activity                -0.0111      0.005     -2.069      0.039      -0.022     -0.001
novelty:activity              -0.0041      0.005     -0.857      0.392      -0.013      0.005
power:novelty:activity         0.0001   8.47e-05      1.258      0.209   -5.98e-05      0.000
valence                        0.8354      0.298      2.803      0.005       0.250      1.420
power:valence                 -0.0237      0.005     -4.783      0.000      -0.033     -0.014
novelty:valence               -0.0114      0.005     -2.129      0.034      -0.022     -0.001
power:novelty:valence          0.0003   8.53e-05      3.117      0.002    9.84e-05      0.000
activity:valence              -0.0092      0.004     -2.267      0.024      -0.017     -0.001
power:activity:valence         0.0003   7.01e-05      4.084      0.000       0.000      0.000
novelty:activity:valence       7.32e-05   7.51e-05      0.975      0.330   -7.42e-05      0.000
power:novelty:activity:valence -2.718e-06   1.29e-06     -2.115      0.035   -5.24e-06 -1.95e-07
==============================================================================
Omnibus:                       14.927   Durbin-Watson:                   1.437
Prob(Omnibus):                  0.001   Jarque-Bera (JB):               15.538
Skew:                           0.353   Prob(JB):                     0.000423
Kurtosis:                       2.843   Cond. No.                     2.14e+08
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 2.14e+08. This might indicate that there are
strong multicollinearity or other numerical problems.
```

E.2. Context to Modality

```
>>> confm
array([[222,  58,  28,   8],
       [ 51, 268,  71,   7],
       [ 28,  54, 127,   9],
       [ 13,   9,   6,  16]])

>>> confm_with_true_sit
array([[265,  33,  18,   0],
       [ 26, 325,  44,   2],
       [ 24,  43, 141,  10],
       [  7,   3,   8,  26]])

>>> confm_with_true_emo
array([[273,  24,  19,   0],
       [ 25, 334,  34,   4],
       [ 24,  43, 144,   7],
       [  7,   6,   3,  28]])

[confusion matrix for true all unfortunately lost]

>>> confm_with_ivec
array([[263,  36,  17,   0],
       [ 21, 329,  43,   4],
       [ 24,  45, 142,   7],
       [  9,   5,   4,  26]])

>>> confm_with_ivec_inf_single_val_act
array([[265,  32,  18,   1],
       [ 24, 339,  30,   4],
       [ 24,  43, 143,   8],
       [  7,   4,   5,  28]])
```

```
 >>> confm_with_ivec_inf_tenfold_sit
 array([[267,  34,  15,   0],
        [ 25, 331,  37,   4],
        [ 25,  41, 145,   7],
        [  9,   5,   3,  27]])

>>> confm_with_ivec_turn
array([[260,  34,  22,   0],
        [ 22, 338,  32,   5],
        [ 25,  38, 147,   8],
        [  8,   5,   5,  26]])

>>> confm_with_ivec_inf_all
array([[269,  31,  16,   0],
        [ 24, 333,  36,   4],
        [ 24,  44, 142,   8],
        [  8,   5,   4,  27]])
```