Human Vocal Event Detection for Realistic Health-care Applications

---

A Dissertation

Presented to

the faculty of the School of Engineering and Applied Science

University of Virginia

---

in partial fulfillment
of the requirements for the degree

Doctor of Philosophy

by

Asif Salekin

August 2019

APPROVAL SHEET

This Dissertation
is submitted in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

Author Signature: Asif Salekin

This Dissertation has been read and approved by the examining committee:

Advisor: John A. Stankovic

Committee Member: Hongning Wang

Committee Member: Yanjun Qi

Committee Member: Yanjun Qi

Committee Member: John Lach, Minor Representative

Committee Member:

Accepted for the School of Engineering and Applied Science:

Craig H. Benson, School of Engineering and Applied Science

August 2019

Abstact

Supported by rapid innovations in machine learning, signal processing, and internet of things technologies, the concept of passive sensing is redesigning almost every aspect of our lives. Innovating novel, low cost and noninvasive sensing techniques to model/identify human events (i.e., emotions, mental disorder, etc.) has become one of the core research interests. Advancement in passive sensing has made the development and operation of complex human health monitoring systems technically feasible. Automated and passive human event sensing can improve assessment and treatment of mental disorders, monitoring and care of patients suffering from agitation, dementia or stroke rehabilitation, extensively reduce the work-load of caregivers, and provide more timely and accurate responses to crisis.

Sound is ubiquitous in the expression of human events and its surrounding environment. According to multiple studies, sound as a modality conveys bio markers of our mental and behavioral states or events. The major scopes of research on human audio event detection are: detection of speech emotion, assessment of mental disorders, behavioral and ambient human event detection. Despite the rapid growth of interest in audio sensing for health applications in recent years, yet, accuracy of detection or modeling human verbal events is far from desirable to have any practical implication. This is due to some open challenges, such as, distortion of acoustic features with variation of speaker to microphone distances, unavailability of strongly labeled audio data, expression of verbal events through consolidation of prosody and context of speech, ambiguity in lexical speech content, limitation of available training data, etc.

In this dissertation, I will present my recent and ongoing research to demonstrate that development and application of novel and adaptive feature engineering approaches, such as, adaptive feature selection, synthetic data generation, and effective feature representation generation, can address the open challenges of human vocal event detection in the scope of health monitoring. With this goal in mind, we have built four automated vocal event detection frameworks that addresses the open challenges in the four major scopes of interest.

Our Distant Emotion Recognition (DER) approach addresses the challenge of acoustic feature distortion due to distance, by a novel distant feature selection approach and a novel, feature modeling/engineering approach, named Emo2vec. A comprehensive evaluation, conducted on two acted

datasets (with artificially generated distance effect) as well as on a new emotional dataset of spontaneous family discussions (38 participants) with audio recorded from multiple microphones placed in different distances, showed presented DER approach achieves a 16% increase on average in accuracy compared to the best baseline method.

This thesis presents a novel weakly supervised learning framework for detecting individuals high in symptoms of mental disorders by addressing the challenge of having weakly (i.e., not well annotated) labeled audio data. Our solution presents a novel feature modeling/engineering technique named NN2Vec to generate low-dimensional, continuous, and meaningful representation of speech from such audio samples, and achieves F-1 scores of 90.1% and 85.44% in detecting speakers high in social anxiety and depression symptoms.

Later, we present DAVE, a comprehensive set of verbal behavioral event detection techniques that includes combining acoustic signal processing with three different text mining paradigms to detect verbal events which need both lexical content and acoustic variations to produce accurate results. Additionally, it adapts a novel word sense disambiguation approach to detect verbal context with multiple ambiguous meanings. Following, the thesis presents a novel framework for ambient human event detection (AHED), which generates robust models for audio monitoring applications with limited available data. The solution presents Audio2Vec, a novel computationally effective feature modeling/engineering technique, and a synthetic training data generation approach from limited audio samples.

Finally, I will discuss the limitations of the presented solutions and lay out my future plans for future improvements.

Acknowledgments

Firstly, I would like to express my sincere gratitude to my advisor Professor John A. Stankovic, who have been an excellent mentor to me. Jack gave me the freedom to explore various research ideas without objection, at the same time guided me through valuable feedback, advice, and encouragement. As an advisor, his constant enthusiasm to try new concepts and ability to contextualize ideas have been instrumental in my growth as a researcher. His advice on both research as well as on my career have been invaluable.

Besides my advisor, I would like to thank the rest of my thesis committee: Dr. Hongning Wang, Dr. Yanjun Qi, Dr. John Lach, and Dr.Yuan Tian. I appreciate their support during my dissertation research, and general advice they provided about academic writing, publishing, career development, and life on the tenure track. I look forward to their continued support and collaboration in publishing, research, and more.

I would like to thank my beloved wife, Merina Jahan. Without her love, support, and patience, I would not have been able to thrive in my doctoral program or balance my research with everything else. I look forward to all the future chapters in life with her.

I would like to express my gratitude to my parents, Habibun Naher and Maminul Haque Sarker, and my sister Nourin Haque Ridi. Without their love and support, I could never have come this far.

Finally, I would like to thank all of my friends, near and far. I am honored to have an eclectic support network to challenge and check in on me. A heartfelt thanks goes out to all of them who provided support, inspiration, mentoring, peer pressure, and motivation along the way.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

According to the U.S. Census Bureaus 2017 National Population Projections [22] by 2030, all baby boomers will be older than age 65. This will expand the size of the older population so that 1 in every 5 residents will be in retirement age. As life expectancy increases, the number of people living with different chronic conditions and functional impairments, for instance, dementia, diabetes and the inability to manage household chores are further increasing.

The Bureau of Labor Statistics Employment Projections [24] for 2016-2026 projected that each year an additional 203,700 new registered nurses will be required. Like the populations they serve, the nursing workforce is also aging. There are currently approximately one million registered nurses older than 50 years, meaning one-third of the nursing workforce could be at retirement age in the next 10 to 15 years [62].

Other than providing healthcare services for the elderly, healthcare and health monitoring at home is also useful for patients who need constant medical care and treatment once they get back to their homes from a medical facility or hospital (i.e., mental disorder, post stroke patients, diabetes, etc.) [161, 144]. There is also a looming shortage of caregivers for continuously monitoring and checking in with large patient populations. And this has brought the adoption of automated human event sensing to the spotlight. A report by the Institute for Health Technology Transformation [70] says, Automation makes population health management feasible, scalable and sustainable. Although, automated human health event sensing and health-assessment cannot replace doctors and nurses, automated sensing and assessment can be blended into their workflows to make a wide swath of care delivery processes much more efficient and to improve productivity. It will extensively reduce the labor time, as a nurse supported by automation tools can handle a larger population of patients at one time.

Passive monitoring via human event sensing takes the notion of resident care, safety, and caregiver monitoring to a whole new level. In passive health monitoring, an automated monitoring system is employed to gather and analyze information on a range of health indicators. Residents do not need

to activate or interact with the sensors in any way. Staff can be entirely hands-off until an alert or detection result is registered. Just as information is collected automatically, automated event monitoring (or detection) system uses it to continuously create a dynamic picture of a residents vital information, analyzing data over time in order to learn the patterns indicative of wellness and those that may signify potential danger.

From the residents point of view, there is a heightened sense of independence and dignity, without sacrificing their health and safety. Residents will experience fewer intrusions into their personal space for check-ins and other wellbeing assessments as the staff can rely on the technical solution for routine monitoring. For the staff, continuous monitoring relieves some of the pressure on any given shift, ensuring an alert will sound if needed, even without constant oversight thus freeing caregivers to allocate their time more effectively.

Moreover, families can obtain peace of mind, as well as the certainty of more open and constant communications. Passive systems can be set to automatically notify loved ones should an event occur, so that families can relax knowing any news that needs to reach them will do so.

However, the biggest win may very well be better clinical outcomes. Remote and passive health monitoring is a system of interlocking technologies, including sensor technology and machine learning. It allows people to continue to stay at home rather than in expensive healthcare facilities such as hospitals or assisted living homes. The system provides more timely responses to crises. It thus provides an efficient and cost-effective alternative to on-site clinical monitoring.

Many of the existing health monitoring systems work based on video information. Video monitoring systems are not robust against conditions, such as, low visible conditions, or obstacles. Moreover, due to their invasiveness they are often not appropriate in private settings. While thermal infrared sensors can be a less invasive alternative, this technology is highly dependent on temperature, and the separation between background and foreground objects can be problematic. In contrast, audio as a monitoring or event detection modality has the following advantages: (1) needs fewer memory storage and computational requirements compared to video streams, hence it is more appropriate for executing on resource constrained devices; (2) unlike cameras, microphones are omnidirectional with no angular limitations; (3) audio event detection is robust against many environmental obstacles; (4) audio as a modality is robust against illumination and temperature; (5) many events have distinctive audio signatures, but little or no video counterpart; and (6) audio-based monitoring systems that perform all the computations locally are potentially more privacy friendly than video.

Speech is the most natural and fundamental means of communication that we (humans) use every day to exchange information. Research shows the human ability to perceive nuances in voices is extremely sophisticated [174]. It may have offered a strong evolutionary advantage, helping our ancestors distinguish familiar from unfamiliar voices, and perceive expressions of need and distress that helped ensure survival. One interesting example can be the visceral reaction we have towards a baby crying: Mothers are even more attuned to their own babys cry, especially if they have given

natural birth [184].

In fact, vocal emotion recognition even has a separate brain region from facial recognition of emotion [168]. When two people talk and truly understand each other, something quite spectacular happens: their brains literally synchronize [180]. It is as if they are dancing in parallel, the listeners brain activity mirroring that of the speaker with a short delay. This is an indicator that our vocal speech and sounds convey bio markers of our mental and behavioral states or events.

In recent years smart technologies such as smart homes, smart cars, home health monitoring and surveillance systems, etc., have become popular among consumers [145, 123]. The mass adoption of artificial intelligence in users everyday lives is also fueling the shift towards voice monitoring. The number of IoT devices such as smart thermostats and speakers are giving voice assistants more utility in a connected users life. Smart speakers, such as, alexa, google home, come with built in microphones. In most modern cars a microphone already exists in the cabin; however, it only starts there. Many industry experts even predict that nearly every application will integrate voice technology in some way in the next 5 years [13]. Hence, effective vocal event detection and assessment has a significant importance in passive home health monitoring.

1.1 Motivation

Interest in research on human audio event detection can be divided into four scopes: emotion recognition, mental disorder detection, behavioral vocal event detection, and ambient human event detection.

Emotion Recognition Emotion is defined, in everyday speech, as a relatively brief conscious experience characterized by a high degree of pleasure or displeasure and an intense mental activity [23]. In human interactions, information is exchanged in many ways such as body language, speech and facial expressions. Much information is implicitly interpreted from speech when people exchange information. The emotional state of a speaker is closely related to this implicit information. It may be expressed or perceived in the volume, speed and intonation of the voice. Both positive and negative emotions in our daily life directly affect our mental or emotional health [108]. Hence, emotion is a fundamental component of health. It is used when investigating people with depression, dementia, cancer, diabetes, obesity, alcoholism, and a myriad of other medical conditions.

Human emotion detection from acoustic speech signals hve significant potential due to its non-intrusive nature (compared to wearables) and pervasive reachability to sensors (compared to video based emotion recognition). Hence, in recent years speech emotion detection is receiving attention with progress of advanced human-computer interaction systems [136, 139]. Also, emotion detection has paramount importance in the entertainment industry, either for the development of emotionally responsive games or toys [76] or for the development of serious games for aiding people with problems to understand social signs [18, 58]. Additionally some potential use of speech emotion detection can

be in smart homes, e-learning [34], smart vehicles [99], etc.

In almost all existing solutions, it is assumed that the individual is next to a microphone in an environment with limited ambient noise. However, this severely restricts the monitoring time. As smart space technologies are developed there is a potential to monitor the individual at all times that they are within rooms that have microphones. This increased knowledge of an individuals' emotional state will significantly improve healthcare for these individuals. For this to work, the speech processing solutions must be able to handle the differences that occur in speech due to various distances to a microphone, with ambient noise, with overlapped conversations, with different reverberations of sound caused by room construction and furnishings, and with other realism found in the wild.

Mental Disorder Detection Approximately 1 in 5 adults in the U.S. (46.6 million) experience mental disorders in a given year [8]. Over one-third (37%) of students with a mental health condition age $14-21$ and older who are served by special education drop out $-$ the highest dropout rate of any disability group [126]. And only 41% of these people who had a mental disorder in the past year received professional health care or other services [109].

Current assessments for *mental disorders* such as social anxiety or depression are based on client self-report and clinical judgment and, therefore, are subject to subjective biases, burdensome to administer, and inaccessible to clients who are not motivated to visit a clinician. Clinician rating scales (e.g., Hamilton Rating Scale for Depression [64]) require training, practice, and certification for inter-rater reliability [118], and client self-reports (e.g., Social Interaction Anxiety Scale ,SIAS [105]) rely on clients' ability and willingness to communicate their thoughts, feelings, and behaviors when distressed or impaired, which can alter their ability and motivation to self-report [14]. Further, distress from these disorders is often difficult for others to detect. For example, socially anxious people rate their own social performance more critically than non-anxious people, even though their actual performance is not necessarily poorer [9, 149, 182]. This suggests that mental disorders can be salient to the person, but not evident to others. Socially anxious people's social avoidance and safety behavior to reduce or hide their anxiety [203] also can limit others' knowledge of their distress. Thus, relying only on subjective approaches for assessment is inadequate for reliable diagnosis, which is problematic given the high prevalence of social anxiety and depression and the vast numbers who receive no help [37]. In the United States, 50% of people with social anxiety and 22% of people with depression never talk with a provider about their symptoms [196]. Moreover, general practitioners correctly identify social anxiety and depression in only 24% and 50% of true cases [115, 202].

Health-care providers would benefit from objective indicators of mental disorders (i.e., automated assessment system), such as, social anxiety and depression symptoms that require no extensive equipment and are readily accessible and not intrusive or burdensome to complement their self-report, interview, and other assessment modalities. Indicators of social anxiety and depression symptoms could improve diagnostic clarity and treatment planning, thereby helping ensure that

people receive the most appropriate interventions. Moreover, automated symptom indicator systems that providers can assess remotely could help close the treatment gap [81] by identifying individuals who may be in need of prevention, assessment, or treatment resources, which could be delivered in person or via eHealth modalities [188]. People with social anxiety may otherwise not seek treatment because, for example, they do not know where to find it, or fear discussing their symptoms with providers [127], and people with depression may not seek treatment in part because they think they can handle or treat their symptoms on their own or do not view their symptoms as pathological [45]. Furthermore, the ability to remotely detect affective states would help providers monitor instances of high affect both between sessions and after the end of treatment. The latter is especially important given the high relapse rates for formerly depressed individuals. Such passive outcome monitoring (e.g., [72]), requiring minimal effort from the client, could help providers identify when the client is distressed and might benefit from a just-in-time intervention or prompt providers and clients to consider scheduling a booster session.

Behavioral and Ambient Vocal Event Detection Understanding and detection of human behavioral events are central to many domains of human endeavor. They offer a window into decoding how one is thinking and feeling. Speech and spoken language communication cues offer an important means for measuring and modeling human behavior. Observational research and practice across a variety of domains from commerce to healthcare rely on speech and language based information for crucial assessment and diagnostic information and for planning and tracking response to an intervention.

For example, studies have shown that approximately 30% -50% of patients with cognitive disorder (dementia) suffer from various forms of agitation [36, 133]. Most of the assisted living facilities rely on the nursing staff and caregivers to monitor and record actions of their patients, and one of the most important events to monitor is agitation. The medical community has defined the Cohen-Mansfield Agitation Inventory [35] which specifies a number of behavioral events for identifying whether a person is suffering from agitation. Five of the most important behavioral vocal events included in Cohen-Mansfield Agitation Inventory [35] are: cursing, constant unwarranted request for help, making verbal sexual advances, asking constant questions and talking with repetitive sentences. Additionally, Cohen-Mansfield Agitation Inventory [35] includes ambient human events, such as, crying, screaming, glass-breaking, etc. Automated detection of these events would help automated assessment and monitoring of patients suffering from agitation.

Human behavioral vocal events, such as, asking for help, cursing, etc, are conveyed through their speech signal, especially through two components: context of speech and tone of speech. Hence, acoustic analysis is significant, since studies [213] have shown that, human behaviors are consistent with specific conscious and unconscious emotion concepts. But, relying only on acoustic signal processing might result in inaccuracy since events such as, asking for help, verbal sexual advances, etc rely heavily on semantics of speech data. Hence, effective extraction and use of both semantic

and acoustic information is important to detect behavioral vocal events.

Human ambient events such as baby crying, screaming, etc. can be useful in variety of applications such as automated health and home monitoring. Though there are some available datasets [137, 52, 119, 120] for ambient vocal events, the amount of labelled event data is significantly low. Limited dataset in training leads to lack of robustness of detection approach in various different environments, which is fundamental requirement for automated audio event monitoring systems.

1.2 Challenges

This thesis addresses a number of key technical challenges towards solving human vocal event detection, specifically in the scope of passive health monitoring.

Distant Emotion Recognition It is difficult to define emotion objectively, as it is an individual mental state that arises spontaneously rather than through conscious effort. Therefore, there is no common objective definition and agreement on the term emotion. This is a fundamental hurdle to overcome in this research area [169]. Additionally, diversity in the way different people speak and express emotions, accents, and age [155] make the task more difficult.

In a realistic indoor speech emotion recognition system, the acoustic sensors, i.e., microphones, capture speech signals originating from sources (humans) situated at various distances. Increasing source-to-microphone distance reduces signal-to-noise ratio and induces noise and reverberation effects in the captured speech signal, thus degrading the quality of captured speech, and hence the performance of the emotion recognizer. Hence, the system has to be robust against de-amplification of the signal due to variable distances, noise and reverberation. We formally call this problem a Distant Emotion Recognition (DER) problem.

Mental Disorders: Difficult to Strong Label Audio Data Studies have shown that prosodic, articulatory, and acoustic features of speech can be indicative of disorders such as depression and social anxiety [97, 176, 38, 51, 98, 167, 195, 175], and research on the objective detection and monitoring of mental disorders based on measurable behavioral signals such as speech audio is proliferating [55, 40, 39, 57, 181]. State-of-the-art works on detecting mental disorders or emotional states (e.g., anxious vs. calm) from audio data use supervised learning approaches, which must be "trained" from examples of the sound to be detected. In general, learning such classifiers requires annotated data, where the segments of audio containing the desired vocal event and the segments not containing that event are clearly indicated. We refer to such data as "strongly labeled." However, diagnosing mental disorders is a complicated and time consuming procedure that requires an annotator with a high degree of clinical training. In addition, strong labeling of mental disorders in speech audio clips is impractical because it is impossible to identify with high confidence which regions of a conversation or long speech are indicative of disorder. Supervised learning, hence, is a difficult task.

Human events dependant on both prosody and semantic of speech In detection of behavioral vocal events from speech two factors are important: the choice of words and acoustic variation. When a speaker expresses a vocal event while adhering to an inconspicuous intonation pattern, listeners can nevertheless perceive the information through the lexical content (i.e. words). On the other hand, some verbal event conveying sentence structures share the same lexical representation with other general statements. If we try to detect a behavioral vocal event, such as, asking for help using only textual features (e.g., using similarity based text analysis and content matching), we can mistakenly identify a story about helping a kid or a discussion about helping others as asking for help. On the other hand, relying only on acoustic signal processing (e.g., temporal pattern mining in the acoustic signal) cannot recognize the situation where people do not depict any specific verbal tone while asking for help, i.e., one might ask for help in a submissive tone or in a dominant tone based on his/her mood.

Hence detection of behavioral vocal events (e.g., asking for help or verbal sexual advances) using only textual inference or only acoustic features results in high false positives and false negatives.

Ambiguity in semantic understanding of human vocal words Some vocal events are perceived from the information through the lexical content of speech. For instance, human vocal event: cursing can be detected from the transcribed text of the speech. But, some words have multiple meanings and only a subset of those meanings may indicate our targeted vocal event. Such as word: 'dog' can be used to describe a pet, also, it can be used as a curse word.

Limitation of Human Audio Event Data Robust human vocal event detection models have to perform well in various environments not introduced in the training phase. Also, input audio signal to noise ratio (SNR) can be very low due to variable source to microphone distances and presence of other ambient noise sources. Training a supervised model robust against unknown environments and low SNR requires sufficiently large dataset with variation of ambient sounds and signal to noise ratios.

Diagnosing mental disorders is a complicated and time-consuming procedure that that requires an annotator with a high degree of clinical training. Moreover due to the presence of personal and sensitive information in spontaneous speech data, oftentimes it is difficult to recruit real patients in such studies. Therefore, available training data for mental disorder detection is very limited.

Similarly, collecting ambient human events, such as, gunshot or baby cry in many different environments or situations is not feasible. There is no existing ambient human event dataset with large variations of background environmental sounds.

Additionally, semantic understanding or extracting textual features for behavioral vocal events, such as, cursing, asking for help requires contextual (text) labeled datasets. There are no existing such datasets.

Hence we need human audio event detection approaches that generate robust models with limited available data.

1.3 Feature Engineering or Modeling

Feature engineering, also known as feature modeling, is the process of constructing new feature representation from existing data to train a machine learning model. Feature engineering is about creating new input feature representation from the existing ones. Feature engineering isolates and highlights key information, which helps the classification algorithms 'focus' on what's important. This step can be more important than the actual model used because a machine learning algorithm only learns from the data we give it, and creating features that are relevant to a task is absolutely crucial [42]. The feature engineering efforts mainly have three goals:

- Preparing the proper input dataset, compatible with the machine learning algorithm requirements.
- Generating effective feature representations highlighting and emphasizing key information that makes classification task easier.
- Improving the performance of machine learning models.

1.4 Thesis Statement

This dissertation investigates the following hypothesis:

By applying novel and application adaptive feature engineering approaches, such as, adaptive feature selection, synthetic data generation, and effective feature representation generation, we can address the open challenges of human vocal event detection and significantly improve system accuracy in realistic health-care and health monitoring applications.

1.5 Contributions

The main contributions of this dissertation are the following:

- The majority of the speech features typically used in vocal event classifiers significantly distort with increase of speaker to microphone distances. Hence, using these features complicate and deteriorate the ability to detect the verbal audio event across variable distances. To address this challenge of feature distortion due to variable speaker distance, noise and reverberation, we develop a **distant feature selection** approach to identify features not significantly affected by distance. Since these features are robust across distance, their distortion with distance is minimal, hence the reduction of accuracy due to distance is reduced (section 3.1). Our evaluation in section 3.4.1 shows, the elimination of distorted features improves 12.1% accuracy and 14% recall for distant emotion recognition compared to the best baseline state-of-the-art solution.

- The modeling/engineering stage of an speech analysis system develops a representation of the speech that reflects the speech information for that specific task. Each small segment of speech represents a state, and audio event is represented by the progression of speech through various states. This dissertation presents a novel **Emo2vec** feature modeling/engineering approach (section 3.2) that assigns a similar vector to the small frames (speech states), which appear in a similar context for a specific emotion. The vectors from the small frames represent the states of speech from the small segments. This representation of states makes differentiation between the states indicative to a specific emotion from the other emotions easier. According to the evaluation in section 3.4.1, Emo2vec approach achieves 10.55% higher accuracy and 11.07 % higher recall for distant emotion recognition, compared to the baseline feature modeling approach: word2vec model.

- A novel weakly supervised learning framework is created for detecting individuals high in symptoms of mental disorders (i.e., social anxiety and depression) from weakly labeled audio data, adding a practical complement to health-care providers' assessment modalities. To our knowledge, no previous research has identified individuals high in symptoms of a mental disorder from weakly labeled audio data.

- A novel feature modeling/engineering technique named **NN2Vec** (section 4.1.3) to generate low-dimensional, continuous, and meaningful representation of speech from long weakly labeled audio data. NN2Vec identifies and exploits the inherent relationship between audio states and targeted vocal events. Identifying individuals high in mental disorder symptoms using NN2Vec achieved on average F-1 scores 15% higher, than those of the other baseline feature modeling techniques (sections 4.4.1 and 4.5).

- A novel multiple instance learning (MIL) adaptation of bidirectional long short term memory classifier, named **BLSTM-MIL** (section 4.2.1) is developed. It is a sequential deep neural network solution that comprehends the temporal properties in speech while also being adaptive to noise in weakly labeled long audio data. Identifying individuals high in mental disorder symptoms using BLSTM-MIL (with NN2Vec features) achieved on average F-1 scores 6.8% higher, than those of the other baseline classifiers (using NN2Vec features). (section 4.4 & 4.5)

- There is no existing dataset that contains spontaneous speech labeled with speakers high in social anxiety. To evaluate our weakly supervised learning framework, we built a dataset consisting of 3-minute samples of weakly labeled spontaneous speech from 101 participants. The study was conducted under the supervision of a licensed clinical psychologist and researcher with expertise in anxiety disorders. A total of 101 participants ranging from 17 to 18 years of age (M = 19.24, SD = 1.84) completed the study in exchange for course credit or payment. Our approach (i.e., BLSTM-MIL classifier with NN2Vec features) achieves an F-1 score of

90.1% in detecting speakers high in social anxiety symptoms, that is 20% higher compared to the best baseline solution. (section 4.4)

- DAVE, an automatic and comprehensive set of techniques (section 5.1) was developed for detecting 5 behavioral vocal events based on both extending various algorithms and combining acoustic signal processing with three different text mining paradigms. We have evaluated DAVE on 34 real agitated elderly (age varies from 63 to 98 years) dementia patients across 16 different nursing homes. We also solve the challenge that dementia patients mumble, speak in low volume and don't articulate words well. (Section 5.3).

- None of the previous state of the art studies has addressed the verbal events: asking for help and verbal sexual advances. We are the first to show that detection of these two vocal events depends both on the acoustic signal and the semantics of the speech. We present a new approach to detect these behavioral vocal events dependant on both prosody and semantic of speech using combination of the acoustic signal processing features and statistical text data mining techniques. Using such a combined feature set we achieve a detection accuracy of 93.45% for asking for help and a detection accuracy of 91.69% for verbal sexual advances.

- Design and implement a modified version of the adapted Lesk algorithm (section 5.1.2) which considers a word's sense, to detect curse words with multiple ambiguous meanings. Using this approach, we have detected cursing with 31% higher accuracy compared to the baseline solution. (section 5.2.3)

- To address the challenge of having small available audio event datasets, we develop an **audio mixture synthesizer**, that generates a large synthetic mixture of labeled isolated audio event clips and various environmental audio clips. Using this automated generalized approach, it is possible to generate any number of well labeled positive and negative synthetic data samples (this can be applied to any audio event with a small available dataset). (section 6.1)

- A novel computationally effective feature modeling/engineering technique, named **Audio2Vec** (section 6.2.1 & 6.2.2) was invented. The generated representations by Audio2Vec are robust against environmental noise, reverberation, and de-amplification of sound due to distance. Moreover, it identifies and exploits the inherent relation between audio states and targeted audio events. As a result, Audio2Vec features can be used with much shallower (less layers & network parameters) neural network classifiers, and achieves significantly higher accuracy compared to the baseline feature representations typically used with much deeper neural networks. Also, shallow networks (for classification) have less execution time which makes them more suitable for real-time audio event detection systems on resource constrained devices. Using the Audio2Vec feature representation we achieve on average 10.3% higher F_1 score compared to the best baseline approach for automated ambient human events detection (section 6.3.2).

1.6 Organization of the Dissertation

The rest of the dissertation is organized as follows:

- Chapter 2 presents the state-of-the-art in technologies related to human vocal event detection.
- Chapter 3 presents a novel solution for Distant Emotion Recognition, addressing the key challenges by identification and deletion of features from consideration which are significantly distorted by distance, creating a novel, called Emo2vec, feature modeling/engineering and overlapping speech filtering technique, and the use of an sequential classifier to capture the temporal dynamics of speech states found in emotions.
- Chapter 4 presents a novel weakly supervised learning framework for detecting individuals high in symptoms of mental disorders. Moreover, it presents a novel feature modeling/engineering technique named NN2Vec (section 4.1.3) to generate low-dimensional, continuous, and meaningful representation of speech from long weakly labeled audio data.
- Chapter 5 presents DAVE, a comprehensive set of verbal behavioral event detection techniques that extracts textual features from transcribed speech as well as extracts acoustic signal features from respective speech portion. Both of the textual and acoustic signal features are used to discriminate the verbal behavioral events from others.
- Chapter 6 presents a novel framework for ambient human event detection (AHED), which generates robust models for audio monitoring applications with limited available data. It presents Audio2Vec, a novel computationally effective feature modeling/engineering technique. Moreover, the generated AHED systems are real-time executable on resource constrained devices.
- Chapter 7 concludes the thesis by summarizing the contributions and describing the future work.

Chapter 2

Related Work

This chapter presents the state-of-the-art in technologies related to human vocal event detection.

2.1 Raw Acoustic Features

Human event detection systems from speech can be split into three parts: feature extraction, modeling/engineering, and classification. Several combinations of features have been investigated for vocal event detection. These features can be divided into two groups according to their time span: low-level descriptors (LLDs) are extracted for each small time frame (16-45 ms is typical), such as Mel-frequency cepstral coefficients, energy, zero crossing rate, pitch, spectral centroid, reduced speech, and reduce vowel space [166, 63, 163, 48, 179, 38, 144, 27]. By contrast, high-level descriptors (HLDs), such as the mean, standard deviation, quartile, flatness, or skewness, are computed using the LLDs extracted for the whole audio signal or for an audio segment covering several frames [148, 165, 163, 194, 44, 26].

2.2 Feature Modeling or Engineering

The feature modeling/engineering stage of an emotion recognition system must obtain a representation of the speech that reflects the emotional information. Depending on the features used, different feature modeling/engineering approaches can be found in the literature. When dealing with LLD, different techniques have been borrowed from other speech recognition tasks, such as supervised and unsupervised subspace learning techniques. Many of these modeling techniques apply windowing to the speech. Recent studies on speech signal processing, achieved improvement on accuracy using the i-vector representation of speech [79, 56]. The i-vector extraction, which was originally developed for speaker recognition, consists of two separate stages: UBM state alignment and i-vector computation. The role of UBM state alignment is to identify and cluster the similar

acoustic content, e.g., frames belonging to a phoneme. The purpose of such alignment is to allow the following i-vector computation to be less affected by the phonetic variations between features. However, the existence of noise and channel variation could substantially affect the alignment quality and, therefore, the purity of extracted i-vectors. The i-vector technique estimates the difference between the real data and the average data and with variance of noise, speaker to microphone distance and reverberation, this difference becomes inconsistent. Additionally, majority portions of positive samples in weakly labeled audio data are actually average or noisy data, that I-vector can not differentiate. Hence, I-vector computation performs poorly on weakly labeled data.

Recently the audio-codebook model [129, 150] has been used to represent the audio signal in windows with "audio words" for vocal event detection. Several studies on text (e.g., word2vec) and language model representations [59, 153, 19, 111] have used various structures of shallow neural networks (one or two hidden layers) to model features. This study considers audio-codebook approach and word2vec approach as baseline feature modeling techniques.

2.3 Emotion Recognition

Various types of classifiers have been used for speech emotion detection, including hidden Markov models [92], Gaussian mixture models [216], support vector machines (SVM) [10], k-nearest neighbor [152] and many others [146].

Recently, deep learning has revolutionized the field of speech recognition. A deep learning approach called, 'Deep Speech' [65] has significantly outperformed the state-of-the-art commercial speech recognition systems, such as Google Speech API and Apple Dictation. With the Deep Learning breakthrough in speech recognition a number of studies have emerged where Deep Learning is used for speech emotion recognition. Linlin Chao [25] et al. use Autoencoders, which is the simplest form of DNN. Another form of Deep Neural Networks is the Deep Belief Networks, which use stacked Restricted Boltzmann Machines (RBMs) to form a deep architecture. [25] tested DBNs for feature learning and classification and also in combination with other classifiers (while using DBNs for learning features only) like k-Nearest Neighbour (kNN), Support Vector Machine (SVM) and others, which are widely used for classification.

With success in speaker ID and speech transcription using i-vector and deep learning, a study [215] used a combination of prosodic acoustic features and the i-vector features and used Recurrent Neural Network to detect speech emotion. We use this solution as one of the baselines. A recent study [191] proposed a solution to the problem of *context-aware* emotional relevant feature extraction, by combining Convolutional Neural Networks (CNNs) with LSTM networks, in order to automatically learn the best representation of the speech signal directly from the raw time representation. CNNs are mostly used in image recognition. This is because, when dealing with high-dimensional inputs such as images, it is impractical to connect neurons to all neurons in the previous volume because

such network architecture does not take the spatial structure of the data into account. Convolutional networks exploit spatially local correlation by enforcing a local connectivity pattern between neurons of adjacent layers: each neuron is connected to only a small region of the input volume. Hence, CNNs are mostly applicable to high-dimensional data. Furthermore, these papers do not consider speaker distance, reverberation or noisy speech. We used this solution as one of our baselines.

Though there is no existing work which addresses the Distant Emotion Recognition (DER) problem, however there are a few works on emotion detection from noisy speech [147, 212, 187]. [147] claims that a work is in progress for emotion detection from noisy speech without any details of their approach. [187] used $CFSSubsetEval$ with $bestfirst$ search strategy feature selection technique to identify features with high correlation with the class, but low correlation among themselves. We use the solution of [187] as one of our baselines.

OpenSMILE is the most extensively used open source feature extraction toolkit. We extract 6552 features, as 39 functionals of 56 acoustic low-level descriptors (LLD) related to energy, pitch, spectral, cepstral, mel-frequency and voice quality, and corresponding first and second order delta regression coefficients according to the most recent INTERSPEECH Computational Paralinguistic Challenge baseline set [170] with the openSMILE toolkit [49]. We train SVM classifiers taking these features as input for emotion recognition (according to [170]). These SVM classifiers with the INTERSPEECH 13 feature set is one of our baselines.

2.4 Mental Disorder Detection

With the recent success of deep learning approaches in speech recognition [65], research on audio event detection studies is shifting from conventional methods to modern deep learning techniques [25, 90]. Several studies [73, 156, 80] have used the convolutional neural network (CNN) to identify the presence of vocal events. The CNN learns filters that are shifted in both time and frequency. Using these filters, the CNN exploits spatially local correlation by enforcing a local connectivity pattern between neurons of adjacent layers: Each neuron is connected to only a small region of the input volume. However, conventional CNNs fail to capture long temporal context information. We consider the CNN as another baseline.

Several studies on detecting emotion and mental disorder from speech [158, 215, 91, 100] have shown that temporal properties in a speech signal provide important information about emotion and mental disorder and have used sequential classifiers such as the recurrent neural network (RNN) and the long short term memory classifier (LSTM). Both the RNN and the LSTM have feedback loops that let them maintain information in "memory" over time. But the LSTM outperforms the RNN [30], as it does better at avoiding the vanishing gradient problem and captures longer temporal context information. Given that LSTM performs better for long audio data, we consider the bidirectional LSTM as a baseline.

Some recent studies [100, 95] have combined a CNN feature extraction architecture with sequential LSTM, an approach named CNN-LSTM that can learn to recognize and synthesize sequential dynamics in speech. In the CNN-LSTM, the CNN acts as the trainable feature detector for the spatial signal. It learns features that operate on a static spatial input (windows) while the LSTM receives a sequence of high-level representations to generate a description of the content. We consider the CNN-LSTM as a baseline.

Due to the presence of label noise (discussed in section 4.2.1), conventional neural networks fall short of generating an optimal solution when trained on weakly labeled data. To generate a deep neural network solution that captures the temporal properties in speech while also being adaptive to noise, we developed a MIL adaptation of BLSTM (BLSTM-MIL). Although no study has used weakly supervised learning to identify vocal events in weakly labeled speech data, several recent studies [86, 75, 183, 87] have detected rare environmental sound events (e.g., car horn, gun shot, glass break) from weakly labeled audio clips (where the event is a small fraction of a long environmental audio clip). Two of these studies [86, 87] used MIL approaches, the mi-SVM and the deep neural network-based MIL (DNN-MIL), for environmental audio event detection. We consider the mi-SVM and the DNN-MIL as baselines (section 4.4.1 and table **??**).

Some recent studies on social anxiety and depression monitoring systems [21, 29, 189] have used smartphone sensors, text information, call information, and GPS data to understand how depression or social anxiety levels are associated with an individual's mobility and communication patterns. To our knowledge, no study has identified whether a speaker belongs to a high versus low social anxiety group or is experiencing an anxious versus calm vocal state from audio data. Although two prior studies [201, 200] have found a strong correlation between vocal pitch (F0) and social anxiety and one study [89] has shown that pitch (F0) and energy are indicative of state anxiety, we used pitch and energy as two of our LLDs.

No study on the detection of depression from speech audio signals [55, 40, 39, 57, 181] has applied weakly supervised learning approaches to weakly labeled audio data. These studies have used LLDs [57, 181] such as pitch, RMS, MFCC, and HNR as features and SVMs [122, 181], hidden Markov models [40], and linear support-vector regression models [57] as supervised learning classifiers. A depression detection approach evaluated on the DAIC-WOZ database [193] used I-vector features with an SVM classifier. Recently, another depression detection study [100] named DepAudioNet evaluated on this database applied a CNN-LSTM classifier using LLD. We consider the two most recent depression detection approaches evaluated on the DAIC-WOZ dataset [122, 100] as baselines.

This study considers the I-vector, audio-codebook, and Emo2vec feature modeling techniques as baselines for comparison (sections 4.4.1 and 4.5).

2.5 Behavioral Vocal Events

Previous works on questions detection considered both textual features and acoustic features. Since, verbal questions detection is a language specific problem several studies from different languages have explored different acoustic features. Pitch, energy, duration and the fundamental frequency have been explored to detect French questions [143, 128], where energy and fundamental frequency were used as features to detect Arabic questions from speech. A recent study [186] has detected English questions with 87.1% precision using pitch, energy and the fundamental frequency.

To utilize the linguistic content of speech some recent studies used textual features in addition to acoustic features for questions detection. [78] combined acoustic features with key words. Unigram, bigram and trigrams, start and end utterance tags, parse tree representation of syntax, etc. have been used as textual features in addition to acoustic features in questions detection from English, French and Vietnamese utterances [142, 20, 103]. A recent study [128] on French questions detection has combined language model features extracted from speech text with acoustic features (duration, energy and pitch) with 75% accuracy.

Bag-of-word features (tf-idf and language model) has not been evaluated for questions detection from English speech utterances. According to our knowledge we are the first to combine acoustic features with unigram and bigram bag-of-word features from speech content to detect questions from English utterances.

According to our knowledge we are the first to detect verbal events: asking for help and verbal sexual advances. There has been a work on English curse detection from twitter data [197], which detects cursing using predefined key (curse) word matching. Hence, this work can not address the challenge of ambiguous curse word detection with multiple meanings. According to our knowledge we are the first to detect curse words with multiple ambiguous meanings from English speech utterances.

There are noninvasive systems used by physicians, that monitor agitated behaviors in dementia patients [116]. There is research that attempts to detect complex agitated behaviors using video data [54]. Also, research has been done for the diagnosis of mild cognitive impairment (MCI) or early dementia using audio-recorded cognitive screening (ARCS) [171]. There is research to detect agitated physical behavior using the skeleton data collected from the Kinect sensor. The focus of that research was to identify agitated activities such as kicking, punching, and pushing [124]. We are the first to detect 5 agitated verbal behaviors from speech and evaluated on real agitated elderly suffering from dementia.

2.6 Environmental Human Events

Several combinations of features and classifiers have been investigated for AHED tasks. From low level audio features, such as Zero Crossing Rate (ZCR) [15], to middle level features, such as Mel-Frequency Cepstral Coefficients (MFCC) and Perceptual Linear Predictive (PLP) [125, 163], and to

high level feature descriptors, such as MPEG-7 [125], these different features and their combination have been used to represent acoustic events.

Clavel et al. [31] propose a Gaussian Mixture Model (GMM) and Maximum A Posteriori (MAP) decision rule-based gunshot detection approach using short-time energy, Mel-Frequency Cepstral Coefficients (MFCC) and spectral statistical moments as features. Vacher et al. [192] also adopt a GMM classifier, with wavelet-based cepstral coefficients as features, for the detection of screams and broken glass. Rouas et al. [154] use MFCC features and a combination of the GMM and Support Vector Machine (SVM) classifiers for detecting screams in outdoor environments. Their method uses an adaptive thresholding on sound intensity for limiting the number of false detections. Sharan et al. [172] evaluates the audio event detection performance of classification techniques for multi-class support vector machines in various noise conditions.

Peter et al. [190] present a car AHED approach using MFCC features that was evaluated on 100 audio clips for each of the events with different signal to noise ratios. Hussein et al. [74] introduce an AHED for smart homes exploring energy, pitch, ZCR, spectral features, MFCC features and K-nearest neighbor (KNN) and SVM classifiers. Nandwana et al. [121], utilized GMM, GMM mean super-vectors and the I-vector framework for an in car AHED. This work was evaluated on 100 audio clips for each of the audio events. Our evaluation considers this I-vector representation as one of the baseline feature representation approaches to which we compare. All these works evaluate on limited acted datasets. Morfi et al. [117] used weakly supervised learning [159] to detect audio events from weakly labeled training data. In contrast, the focus of this study is to use available strong labeled audio data to generate a robust real-time AHED solution.

Recently, research on audio event detection studies are shifting from conventional methods to modern deep neural network approaches [107, 90]. Several studies [185, 88, 130] have used Convolutional Neural Networks (CNN) with a large input field to identify the presence of an audio event in a large audio clip. CNN learns filters that are shifted in both time and frequency, hence can cover a large input field. Our evaluation considers CNN as one of the baseline classifier approaches.

In recurrent neural networks (RNNs), information from previous time steps can in principle circulate indefinitely inside the network through the directed cycles, where the hidden layers also act as memory. Hence, an RNN can capture comparatively long temporal context information. Parascandolo et al. [131] propose an audio tagging approach based on bi-directional long short term memory (BLSTM) and evaluate 60 different sound events detection in 103 real life recording clips. Marchi et al. [102] and Wang et al. [199] also apply an LSTM classifier on limited audio data for AHED. Our evaluation considers a bi-directional LSTM as one of the baseline classifier approaches.

Chapter 3

Distant Emotion Recognition

It is difficult to define emotion objectively, as it is an individual mental state that arises spontaneously rather than through conscious effort. Therefore, there is no common objective definition and agreement on the term emotion. This is a fundamental hurdle to overcome in this research area [169]. Additionally, diversity in the way different people speak and express emotions, accents, and age [155] make the task more difficult. However, when speaker to microphone distance increases (as opposed to when the microphone is right next to the speaker), it adds further complexity to the emotion detection problem due to room reverberation, noise, and de-amplification of speech. A realistic emotion detection system which is deployed in open environments such as homes, captures sound waves from distant sources (human subjects). We formally call this a Distant Emotion Recognition (DER) problem. This chapter addresses the DER problem by presenting new solutions for LLD feature selection and feature work/engineering, as well as using a LSTM classification technique to exploit the temporal information across low-level speech states represented by minimally distorted features.

The contributions of this DER work are:

- The majority of the speech features typically used in vocal event classifiers significantly distort with increase of speaker to microphone distances. Hence, using these features complicate and deteriorate the ability to detect the emotional state across variable distances. In our solution we have identified 48 low-level descriptor features which do not significantly distort across variable speaker to microphones distances. We use these features as core elements of the overall solution.

- As shown in figure 3.1, we segment an audio clip into overlapping small frames and extract these 48 robust low-level descriptor features (LLD) from them. Each small segment of speech represents a state and an emotion is represented by the progression of speech through various states. We develop a novel *Emo2vec* feature modeling/engineering approach (section 3.2) that

assigns a similar vector to the small frames (speech states), which appear in a similar context for a specific emotion. The vectors from the small frames represent the states of speech from the small segments. In addition, we exploit temporal dynamics of these states which provides rich information for speech emotion. The temporal information in our emotion detection approach is used through a long short term memory classifier (LSTM), where the sequence of vectors (from small frames using Emo2vec feature modeling) are used as input.

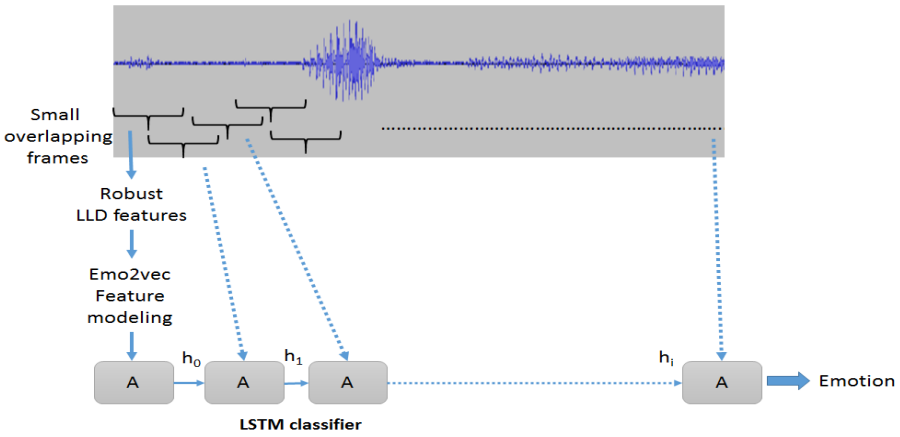


Figure 3.1: Overview of our approach

- Most of the existing approaches to automatic human emotional state analysis are aimed at recognition of emotions on acted speech. A common characteristic of all the existing acted emotional speech datasets is that all of them are made of clean speech recorded by closely situated microphones, often in a noise-proof anechoic sound studios. All the existing speech emotion recognition results are based on these clean speech recordings and, hence, these results are inaccurate in a real world environment where acoustic sensors are likely to be situated far from the speakers. To evaluate our approach on speech with various levels of reverberation and de-amplification (generally due to distance and the environment), we used two acted emotion datasets (section 3.4.1). We trained our model on clean training data and evaluated on disjoint modified test data (by introducing various reverberation and de-amplification effects). Through this evaluation we achieved 90.64%. 89.41% and 90.88% accuracy and 92.7%, 90.66% and 90.86% recall for emotions happy, angry and sad, respectively, which is 10.5%, 9.7% and 10.3% improvement in accuracy (and 10.2%, 12.2% and 15.85% improvement in recall) compared to the best baseline solution. (section 3.4.1)

- The fact that acted behavior differs in audio profile and timing from spontaneous behavior has led research to shift towards the analysis of spontaneous human behavior in natural settings

[214]. There is no existing spontaneous human emotion speech dataset with audio recorded from multiple microphones placed at different distances. Hence, we have built a new emotional dataset of spontaneous family discussions (Collected from 12 families, a total 38 people) where the audio was collected from 3 different distances: 2.25, 3 and 5.3 meters (section 3.4.2). Evaluation of our solution on this dataset shows an average 91.42%, 89.1% and 88.04% accuracy and 91.9%, 89.26% and 86.35% recall for emotions happy, angry and sad, respectively, across various distances (Table 3.6 in section 3.4.2).

- One of the major challenges in the realistic conversations is overlapping of speech, which none of the previous works on speech emotion detection has addressed. This study introduces a novel overlapping speech filtering approach for the DER problem without needing expensive microphone arrays (section 3.4.2) which increases accuracy for happy and angry emotions up to 92.71% and 90.86% (92.43% and 91.21% recall), respectively, across various distances (table 3.7).

- We have implemented 4 baseline solutions from the literature and compared those solutions with our solution on both acted datasets (section 3.4.1) and our newly created spontaneous family discussion dataset (section 3.4.2). According to our evaluation our novel overall DER solution achieves approximately 16% increase in accuracy compared to the best baseline method.

3.1 Challenges and Solution: Feature distortion

With the increase of speaker to microphone distance, recorded signals start to distort compared to the original signal due to de-amplification of the signal, reverberation and ambient noise of the room. The amount of distortion depends on the speaker to microphone distance, the acoustic properties of the room and the amount of noise. To address this challenge our solution is to identify features not significantly affected by distance. Since these features are robust across distance, their distortion with distance is minimal, hence the reduction of accuracy due to distance is reduced. To measure the distortion of a feature across distance d we use the equation 3.1 where f_0 and f_d are the feature values for a clean signal and signal from distance d, respectively.

$$distortion_d = |\frac{f_0 - f_d}{f_0}| \times 100\% \tag{3.1}$$

The following subsections discuss the data collection strategy for our experiment, the extracted features and the identification of robust features from low-level descriptor frames.

3.1.1 Controlled Lab Experiment

We recruited 12 people to read scripts in a controlled lab experiment. We used a VocoPro UHF-8800 Wireless Microphone System and a transmitter, M-Audio Fast Track Ultra 8R USB 2.0, to record and transmit sound. The microphone setting is shown in Figure 3.2. It was a rectangular room and 7 microphones were placed facing the speaker. One was 0.5 meters away, three were about 1.5 meters away, two were about 3 meters away and the last one was about 6 meters away. Multiple microphones were placed at the same distances to record sound from different angles. Each microphone recorded speaker's voice separately, but simultaneously, into 44.1kHz, 32-bit wav format files. Each speaker read 64 scripts (duration ranging from 5 to 20 seconds). All the microphones could record the speech. The purpose of this dataset is to identify robust features across distance not emotion detection.

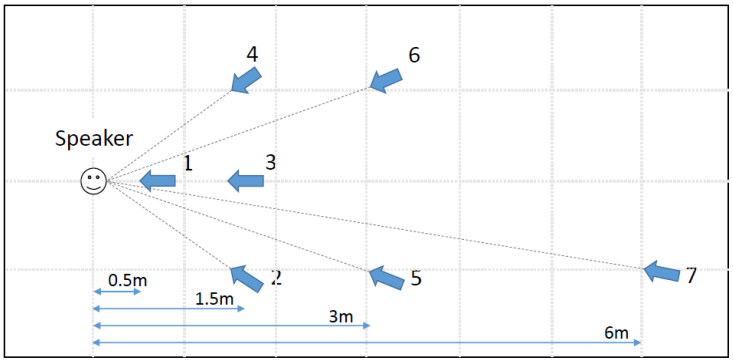


Figure 3.2: Microphone setting in Lab

3.1.2 Select Robust Features

Based on the previous studies on acoustic features associated with emotion (section 2.1) we considered 77 low-level descriptor (LLD) features shown in Table 3.1, as well as their delta and delta-delta coefficients (total 231 features). Collected emotional speech audio clips from our controlled experiment are segmented into 25ms small frames (with 10ms overlapping). 231 LLD features typically used in emotion detection are extracted for each of these 25ms small frames. We calculated distortion considering the 0.5 meter distance microphone audio as clean speech in equation 3.1. Figure 3.3 shows the number of LLD features for various average distortion ranges. According to our evaluation all delta and delta-delta features distort more than 100%. Also, a majority of the rest of the features distort between 40% to 50% when speaker to microphone distance is 6 meters. Hence, through our evaluation we considered 48 LLD features, with less than 50% distortion through various distances to use as attributes in our DER approach. These features are 5 Mel-Frequency

Feature	Count
Mel-Frequency cepstral coefficients (MFCC) 1-25	25
Root-mean-square signal frame energy	1
The voicing probability computed from the ACF	1
The fundamental frequency computed from the Cepstrum	1
Pitch	1
Harmonics to noise ratio (HNR)	1
Zero-crossing rate of time signal	1
PLP cepstral coefficients compute from 26 Mel-frequency bands	6
The 8 line spectral pair frequencies computed from 8 LPC coefficients	8
Logarithmic power of Mel-frequency bands 0 - 7	32

Table 3.1: Considered acoustic features associated with emotion

cepstral coefficients, voice probability, fundamental frequency, zero crossing rate, 8 line spectral pair frequencies and 32 Logarithmic power of Mel-frequency bands.

3.2 Feature Modeling/Engineering

The modeling/engineering stage of an speech analysis system develops a representation of the speech that reflects the speech information for that specific task. Each small segment of speech represents a state, and emotion is represented by the progression of speech through various states. In the DER problem we consider a speech signal from small frames (25ms) to represent a state of speech. The following sections introduce a representation of the speech state from a small frame that handles the small distortion of LLD features due to reverberation or speaker to microphone distance variance, as well as takes into account the relationship of a particular state with its neighbor states for each particular emotion.

3.2.1 Audio to word

We use the Audio-Codebook model [129, 150] to represent the audio signal from small frames with ‘words’. These ‘words’ represent the state of speech in the small frames and are not words in the normal meaning attributed to words. In our context the Audio-Codebook words are fragments of

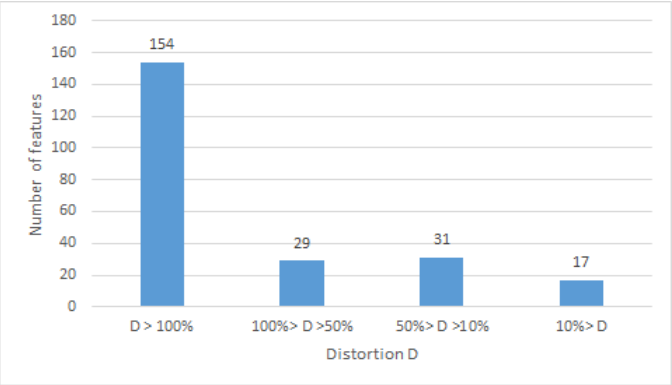


Figure 3.3: Number of LLD features on various average distortion range

speech represented by features. We use the k-means clustering method to generate the audio codebook from the LLD feature representations mentioned in section 3.1.2. K-means is an unsupervised clustering algorithm that tries to minimize the variance between the k clusters and the training data. In the codebook generation step, we first randomly sample points from the audio in the training set and then run k-means clustering. The centroids of the resulting clusters form our codebook words. Once the codebook has been generated, acoustic LLD features within a certain small range of the speech signal are assigned to the closest (Euclidean distance) word in the codebook. As it might be the case that one input frame has a low distance to multiple audio words and, hence, the assignment is ambiguous, we take multiple assignments into account. As shown in the figure 3.4, the N_c nearest words from codebook are assigned to a small frame.

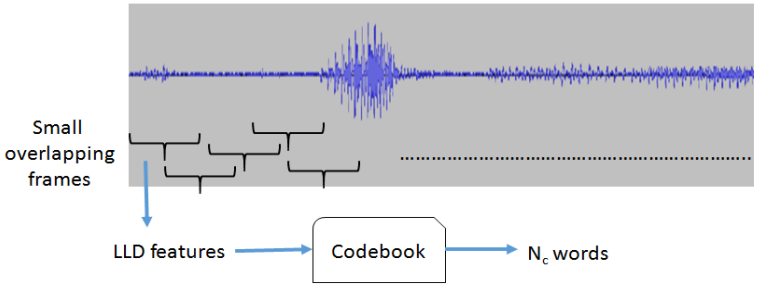


Figure 3.4: Extraction of codebook words

The LLD features selected from section 3.1.2 distort up to a certain threshold with variance of speaker to microphone distance and reverberation. Our trained audio codebook places similar points in the feature space into the same words, which reduces the effect of feature distortion to a certain level.

The discriminating power of an audio codebook model is governed by the codebook size. The

codebook size is determined by the number of clusters K generated by the k-means clustering. In general, larger codebooks are thought to be more discriminative, whereas smaller codebooks should generalize better, especially when LLD features extracted from small frames can distort with distance, noise and reverberation, as smaller codebooks are more robust against incorrect assignments. We have evaluated with different size of codebooks as described in our evaluation section.

3.2.2 Basic Word2vec Model

In this section, we present a brief description of the skip-gram model [59, 153] and in the following section discuss the novel enhancement of this model for our DER solution. The objective of the skip-gram model is to infer word embeddings (vectors) that are relevant for predicting the surrounding words in a sentence or a document, which means if two different words have very similar 'contexts' (i.e., words which appear around them frequently in training), their vectors are similar.

More formally, given a sequence of training words $w_1, w_2, \ldots, w_T$, the objective of the skip-gram model is to maximize the average log probability, shown in equation 3.2

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\leq j\leq c, j\neq 0} logp(w_{t+j}|w_t) \tag{3.2}$$

where c is the size of the training context (which can be a function of the center word w_t). The basic Skip-gram formulation defines $p(w_{t+j}|w_t)$ using the softmax function:

$$p(w_O|w_I) = \frac{exp(v_{w_O}^{'T} v_{wI})}{\sum_{w=1}^{W} exp(v_w^{'T} v_{wI})} \tag{3.3}$$

where v_w and $v_w^{'}$ are the 'input' and 'output' vector representations of w, and W is the number of words in the vocabulary.

For example, if we take two N_v dimensional vectors for two words, that are randomly initialized with some values (shown in figure 3.5a), and if we add a tiny bit of one vector to the other, the vectors get closer to each other, simply by virtue of vector addition. Figure 3.5b shows this for 2 dimensional vectors OA and OB. If we subtract a tiny bit of one vector from the other the vectors move apart by a tiny bit (shown in figure 3.5c). During word2vec training, in each step, every word (vector) is either pulled closer to words (vectors) that it co-occurs with, within a specified window or pushed away from all the other words (vectors) that it does not appear with. Word2vec training only brings together words (vectors) that are within the specified window context.

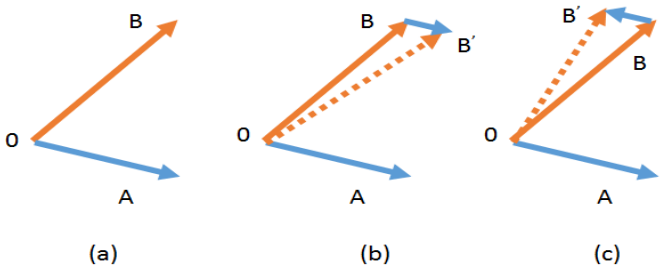


Figure 3.5: Explain word2vec

3.2.3 Novel Emo2vec model

This section introduces a new Speech Emo2vec solution, an acoustic emotion specific word2vec model, which performs more effectively than word2vec (shown in section 3.4.1) for emotion detection. The objective is to generate vectors from the LLD features extracted from small frames. These vectors are the inputs to the classifier. These new vectors for each frame are generated in a manner which indicates that if two frames appear in a similar context (i.e., similar surrounding frames) and for a specific emotion that the vectors will be similar. This representation of states makes differentiation between the states indicative to a specific emotion from the other emotions easier.

According to this model for every N_c words (from codebook) extracted from a small frame f_i, in the vector identification (word2vec) training, the number of neighbor words is $2 \times (N_c \times c)$, extracted from the left and right c windows of the frame f_i. As shown in figure 3.6, if $N_c = 3$ and $c = 4$, for each word $w_{i,1}, w_{i,2}$ or $w_{i,3}$, the neighbor word set in the word2vec training consists of all the words extracted from $i - b$ and $i + b$ small frames, where $b = 1, 2, 3, 4$.

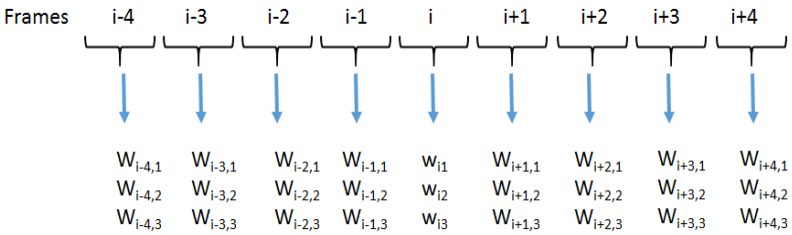


Figure 3.6: Emo2vec: vector calculation using neighbour small frames

To achieve similar vectors for the small frames f_i, which occurs in similar context for a specific emotion E, we train a Emo2vec model for each of the emotions with input corpus D_e. D_e is a collection of $(w, Neighbour_w)$ pairs, that occurs in the training speech signal samples for that specific emotion E. Here, w is a word from a small frame f_i, and $Neighbour_w$ is a set consisting of $2 \times (N_c \times c)$ words extracted from the left and right c windows of the frame f_i. Training Emo2vec,

generates similar vectors for the *words* $\in w$ from corpus D_e, if the *words* have a similar neighbour set $Neighbour_w$ multiple times.

To illustrate further, figure 3.7 shows an example input training corpus for detection of emotion: happy. Here, $N_c = 2$ and $c = 2$, hence each word has 8 neighbours. In this figure the word-neighbor pairs in the left are from happy speech samples and the right are the ones from other speech samples. According to figure 3.7 words A, B and, C have similar neighbours. Word2vec training [94, 209] considers samples from the whole corpus (both D_H and D_N). Hence, generated vectors for words: A, B, and, C would be similar. Since, these words occurs with similar neighbour words and generated Word2vec vectors for them are also similar, differentiation task for happy emotion detection classifier is difficult. But, Emo2vec training only considers happy speech samples (corpus D_H), hence generated Emo2vec vectors for only words A and B are similar. Since, according to Emo2vec vector representation, A and B words are closer in feature space (similar vectors) and further from C, the differentiation task for the classifier is easier. Emo2vec put words which occur in similar context (similar neighbour set) for a specific emotion (in this example happy), closer in feature space. Words which occur with similar neighbours, but for different emotions are pushed further apart. Hence, the classification task for that specific emotion gets easier.

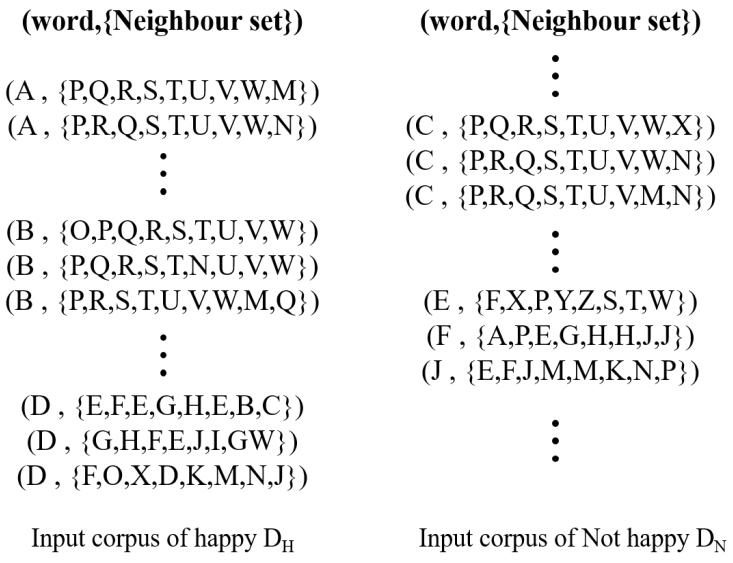


Figure 3.7: Example training corpus for emotion: happy

Additionally, we perform the following steps before training our emotion specific Emo2vec model for each of the emotions:

Sub-sampling Frequent Words:

Sub-sampling is a method of diluting very frequent words. Very frequent words are equivalent to stop words in text documents (is, are, the, etc.). Analogously, there are some codebook *words* (section 3.2.1) which appear in nearly all speech samples of happy, angry and sad emotions. Hence they are not discriminative to any emotion. Our solution deletes these frequent *words* from consideration in our emotion classification.

Before training Emo2Vec models for each of the emotions E_i using training corpus D_i, where i =happy, angry or sad, we sub-sample the frequent *words* which appear more than a threshold t_f times across all training corpuses D_i. The sub-sampling method randomly removes words that are more frequent than some threshold t_f with a probability of p, where f marks the word's corpus frequency:

$p = 1 - \sqrt{\frac{t_f}{f}}$

A modified version of the word2vec [113] performs sub sampling frequent words from the corpus it trains on. Applying that approach would eliminate frequent words appearing for our targeted emotion (for example: happy). But, that frequent word can be rare for other emotions, hence highly indicative to our targeted emotion. Elimination of such highly discriminative words would make classification difficult. Hence, we perform sub sampling frequent words removal across all training corpuses (for all emotions), before performing training of Emo2vec.

Deletion of rare words across all emotions:

If a word w_i appears less than a threshold t_r times across all training corpuses (happy, angry and sad), we delete w_i before creating the context windows. The intuition is, rare words are too specific and too small in number, hence they are indicative to some specific audio clips, rather than a generic class of audio clips, in our case audio from specific emotions.

After sub-sampling frequent words and deletion of rare words, through our emotion specific Emo2vec approach we generate vectors for each of those words occurring between t_r to t_f times in the training corpuses (for all emotions). We name this emotion specific word to vector mappings as the Emo2vec dictionary.

3.2.4 Speech Emo2vec extraction

As shown in figure 3.8 we extract LLD features from each of the small frames $frame_k$. According to section 3.2.1, N_c (solution uses $N_c = 3$) words are extracted for that LLD feature set using the generated codebook. Using the Emo2vec dictionary generated in training phase we convert the words to their corresponding vectors of size s. Unseen words w_j, where Emo2vec dictionary does not contain word to vector mapping, are represented with zero vector of size s. Extracted word vectors are added to create final output vector $V = [v_1, v_2, \ldots, v_s]$ of size s. This output vector V represents

the state of speech signal from small frame $frame_k$.

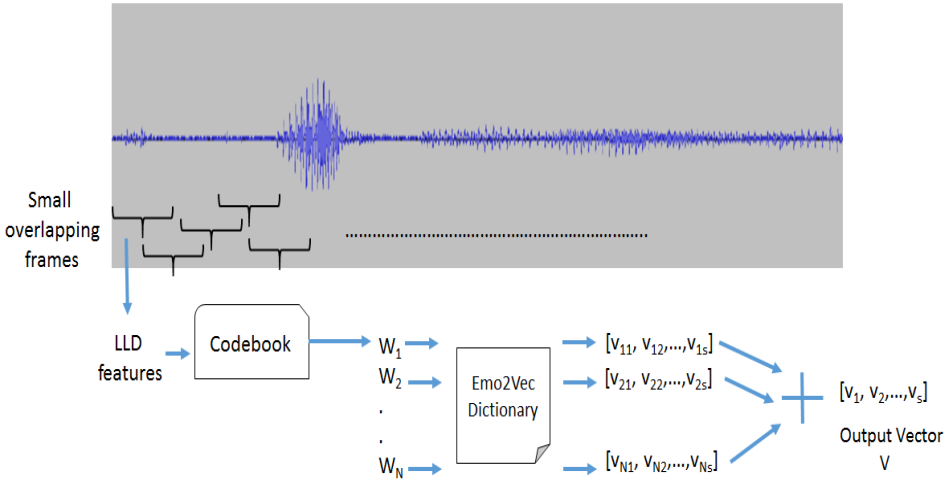


Figure 3.8: Emo2vec vector extraction from speech

3.3 Classification: Recurrent Neural Network and LSTM

Recurrent Neural Networks [112] are a type of Neural Network where the hidden state of one time step is computed by combining the current input with the hidden state of the previous time steps. They can learn from current time step data as well as use knowledge from the past that are relevant to predict outcomes. RNNs are networks with loops in them, allowing information to persist. In the figure 3.9, a chunk of a neural network, A, looks at some input x_t and outputs a value h_t. A loop allows information to be passed from one step of the network to the next. If we unroll the loop, a RNN can be thought of as multiple copies of the same network, each passing a message to a successor. This chain-like nature reveals that recurrent neural networks are intimately related to sequences and lists. They are the natural architecture to use for such data.

Formally, the output of an RNN is computed as:

$$\hat{y}_t = \sigma(W_o h_t) \tag{3.4}$$

Here, W_o is a parameter matrix of the model, h_t is the hidden state vector of the Neural Network, σ is some differentiable function that is applied element-wise and $\hat{y}_t$ is a vector containing the predicted output. If the input is a sequence, $x = x_1, x_2, \ldots, x_t$, the hidden state of a step t is

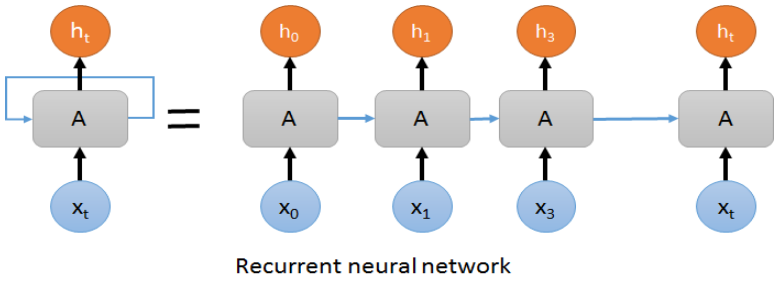


Figure 3.9: Recurrent Neural Network

computed by:

$$h_t = f(h_{t-1}, x_t) \tag{3.5}$$

Since, the memory of RNNs is essentially unlimited, RNNs can in principle learn to remember any length of states. RNNs capture long-term temporal dynamics using time-delayed self-connections and are trained sequentially. In our approach, an audio signal is segmented into overlapping small frames, and an Emo2Vec V vector is extracted from each of these small frames. Emo2vec V represents the state of speech from a small frame (25ms in our solution). Intuitively, it is not possible to perceive speech emotion from a 25ms small frame, but emotions can be recognized by the the temporal dynamics across these states (represented by Emo2vec vector V). Consequently, we use the temporal information in our emotion detection through a recurrent neural network (RNN). The sequence of vectors (Emo2Vec vectors), each of which represents the state of speech from a small overlapping frame, are the input sequence x_i, $i = 0, 1, .., t$ to a RNN.

RNNs can detect the final output (in this case an emotion) using Emo2Vec V vectors as input from any length of small frame sequences. This means that our emotion detection using Emo2vec features is not limited to any fixed window size; it can detect emotion capturing the progression of speech states (represented by Emo2Vec) from any variable size windows.

3.3.1 Long Short-Term Memory Units

A study [132] showed that it is not possible for standard RNNs to capture long-term dependencies from very far in the past due to the vanishing gradient problem. The vanishing gradient problem means that, as we propagate the error through the network to earlier time steps, the gradient of such error with respect to the weights of the network will exponentially decay with the depth of the network. In order to alleviate the gradient vanishing problem, [69] developed a gating mechanism that dictates when and how the hidden state of an RNN has to be updated, which is named as long short-term memory units (LSTM).

The LSTM contains special units called memory blocks in the recurrent hidden layer. The memory blocks contain memory cells with self-connections storing the temporal state of the network

in addition to special multiplicative units called gates to control the flow of information. Each memory block in the original architecture contains an input gate and an output gate. The input gate controls the flow of input activations into the memory cell. The output gate controls the output flow of cell activations into the rest of the network.

Emotion is represented by the progression of speech through various states. Our solution represents a state of speech by a Emo2vec vector. Comprehension of temporal dynamics of states throughout the entire speech segment (that we want to classify) requires long-term dependency. Hence, to avoid vanishing gradient problem, our solution uses the LSTM defined by [60] as our Recurrent Neural Network classifier. The implementation (equations 3.6,3.7,3.8,3.9, and 3.10) adds the additional forget gate, which addressed a weakness of LSTM models preventing them from processing continuous input streams that are not segmented into sub-sequences.

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \tag{3.6}$$

$$r_t = \sigma(W_{xr}x_t + W_{hr}h_{t-1} + W_{cr}c_{t-1} + b_r) \tag{3.7}$$

$$c_t = r_t \odot c_{t-1} + i_t \odot tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \tag{3.8}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o) \tag{3.9}$$

$$h_t = o_t \odot tanh(c_t) \tag{3.10}$$

Here, $\odot$ is the element-wise product and i, r and o are the input, forget and output gates, respectively. As it can be seen, the gating mechanism regulates how the current input and the previous hidden state must be combined to generate the new hidden state.

3.3.2 Dropout Regularization

During the training phase of a neural network, neuron weights settle into their context within the network. Weights of neurons are tuned for specific features providing some specialization. This makes neighboring neurons dependant on this specialization, and with further training it can result in a fragile model too specialized to the training data. This dependency on context for a neuron during training is referred to as complex co-adaptations. If neurons are randomly dropped out of the network during training, neighboring neurons would have to step in and handle the representation required to make predictions or classification for the missing neurons. This is believed to result in multiple independent internal representations being learned by the network. The effect is that

the network becomes less sensitive to the specific weights of neurons. This, in turn, results in a network that is capable of better generalization and is less likely to overfit the training data. This regularization is named as dropout regularization. [178].

Our solution uses a two layer LSTM model. The first layer is the LSTM layer with N_{neuron} neurons. 20% dropout rate is set for this layer, which means two in 10 neurons were randomly excluded from each update cycle. Since this is a classification problem, we use a dense output layer with a single neuron and a sigmoid activation function to make 0 or 1 predictions for the two classes (Emotion E or not emotion E). Log loss is used as the loss function and the efficient gradient descent optimization algorithm is used.

3.4 Evaluation

Our evaluation consists of two parts. First, in section 3.4.1, two existing acted emotion datasets from the literature are used and distance issues are emulated by incorporating artificial reverberation and de-amplification. We use these generated segments to evaluate our approach. This section addresses different questions as well as compares our solution with the state-of-the-art solutions. Second, since, there is no existing spontaneous human emotion speech dataset with audio recorded from multiple microphones placed at different distances, in section 3.4.2 we describe the creation of a new dataset. We recruited 12 families and let them exhibit different emotions in spontaneous unscripted discussions. Multiple microphones were used to record the discussion sessions. Several challenges and our solutions to address them, and comparison with state of the art baselines on this family discussion data are discussed in section 3.4.2.

3.4.1 Experiment on Acted Data

Emotional Datasets

We use two emotion speech datasets: EMA and SAVEE (with annotations for 3 emotions: happy, angry, sad and others) where the spoken language was English and speakers were close to the microphone. The Electromagnetic Articulography (EMA) dataset [93] includes articulatory motions recorded by an EMA system where talkers produced simulated (acted) emotional speech. There were 3 participants: a male and two females who are native speakers of American English. In total, the male participant produced 280 utterances (14 sentences with 5 repetitions for each of the 4 emotions), and each of the female participants produced 200 utterances (10 sentences with 5 repetitions for each of the 4 emotions). The SAVEE dataset [66] contains 15 sentences for each of the 4 emotion categories, from 4 male participants. In this dataset some sentences are natural while others are acted or elicited. In total there are 230 audio clips for each of the 4 categories: Happy, angry, sad, and others. These two datasets were merged for the evaluation.

Acoustic Pre-processing

This section describes the noise filtering, acoustic de-amplification, and reverberation of speech we used in the evaluation of these two datasets.

Filtering and Removing Noise: The first step of pre-processing is to remove unvoiced audio segments using zero crossing rate (ZCR) [16]. To capture the pause in spoken sentences, the detected voiced segments are lengthened by 1 second on both sides. If another consecutive voiced segment starts within the lengthened 1 second segment portion, both the voice segments are merged into one.

Noise which is out of human voice frequency range were removed using a bandpass filter with a low frequency of 80Hz and a high frequency of 3000Hz. Hiss, hum or other steady noises were reduced using a spectral noise gating algorithm [7].

Reverberation and de-amplification Reverberation refers to the way sound waves reflect off various surfaces before reaching the listener's ear. In recent years a few dereverberation methods has been developed, though blind one-microphone dereverberation approaches are not so accurate yet [211]. However, there are different artificial reverberation effect parameters to model how sound waves reflect from various types of room size and characteristics. In this study, we use different combinations of reverberation parameters: wet / dry ratio, diffusion, and decay factor. Wet and dry ratio is the ratio of the reverberated signal to the original signal. The more reverberation the room has, the larger this ratio is. Diffusion is the density rate of the reverberation tail. A higher diffusion rate means the reflection is closer and the sound is thick. A lower diffusion rate has more discrete echoes. Decay factor is used to measure the time duration that reflection runs out of energy. A larger room has longer reverberation tails and lower decay factors. A smaller room has shorter tails and higher decay factors.

De-amplification decreases the loudness or volume levels of the audio clip and produces the effect of increase in speaker to microphone distance.

Training and Data Preparation

We perform filtering and noise removal (section 3.4.1) on the datasets (section 3.4.1) to remove noise and unvoiced audio segments. Our evaluation performs 5-fold cross validation using 90% of the dataset for training and other 10% for testing. In each of the 5 iterations, we train our model on clean, close-to-microphone training data (which is the original recordings in the dataset from section 3.4.1). We apply different combinations of de-amplification along with reverberation parameters: wet / dry ratio, diffusion, and decay factor on the audio clips of the test dataset (remaining 10% of dataset) to artificially introduce the effect of different speaker to microphone distances with different room characteristics. Finally, we evaluate our classifier on this artificially reverberated and de-amplified test data. This experiment measures the robustness of our approach in terms of

reverberation and de-amplification, i.e, if a model is trained using clean (no reverberation), close of microphone audio data, how it performs on data with reverberation with different distant speakers (artificially generated).

Following previous audio-codebook generation approaches [129, 150] we randomly select 30% of the training data from all emotions to generate the audio-codebook. We have trained an individual binary classifier for each of the emotions: happy, angry and sad. To generate emotion specific Emo2vec dictionary our approach randomly selects 50% clips from the training set of that respective emotion.

Results

We next describe the efficiency and the applicability of our solution by investigating some of the pertinent questions related to the DER problem.

What are beneficial parameter configurations? There are a number of parameters in our emotion detection solution. They are sub-sampling parameter t_f (section 3.2.3) which defines the threshold of frequent words, t_r from section 3.2.3 which defines the threshold of rare words and number of neighbour window parameter c (section 3.2.3). Through our evaluation we identify that the beneficial value of t_f is 880, t_r is 4 and c is 4. Also, our two layer LSTM classifier has N_{neuron} neurons in its' first (hidden) layer. Through our evaluation we identify the beneficial value of N_{neuron} is 100 neurons. Identification of the beneficial values were performed through iterating multiple values within a range. For example, we iterate the value of the sub-sampling parameter t_f over a range from 400 to 1000, and identify that using t_f=880 as sub-sampling parameter facilitates higher accuracy for all of our targeted emotion recognitions.

As shown in section 3.2.1, the codebook size influences the discriminative characteristics of our feature modeling/engineering approach. If the codebook size is too large, it will be more discriminative, but lose the reduction capability of feature distortion. Also, a small codebook will be too generic, hence, will not be able to capture the change of emotional speech states in various small frames. Suppose we select codebook size CB (e.g. 1000), if we represent the generated words by one-hot encoded vector of size of vocabulary CB, the number of features from each small frames for our LSTM classifier will be CB. Emo2vec condenses that dimensional vector. Smaller Emo2vec vectors are better for computation and performance on a dataset of limited size, but too small a vector size loses relational and discriminative information between the words.

As shown in table 3.2, we evaluated (according to section 3.4.1) our solution with various codebook and Emo2vec vector sizes. Table 3.2 shows our average 5-fold cross validation results where classifiers were trained on clean, close to microphone data and tested on artificially reverberated and de-amplified data. According to this table, accuracy for emotion happy and angry achieved up to 90.64% and 89.41% (92.7% and 90.66% recall) with codebook size 500 and Emo2vec vector size 50.

Codebook size	Happy Emo2Vec size			Angry Emo2Vec size			Sad Emo2Vec size		
	30	50	100	30	50	100	30	50	100
500	89.4	90.64	90.176	88.8	89.41	89.19	79.61	80.1	81.84
1000	85.9	86.29	89.55	85.2	86.02	88.35	79.1	80.1	83.94
1500	85.9	86.29	88.12	82.9	85.92	87.2	85.33	87.72	90.88
2000	83.2	85.17	87.29	82.9	84.6	86.9	86.72	89.5	85.33

Table 3.2: Accuracy of Emotion detection with various codebook and Emo2vec vector sizes

Emotion	Angry		Happy		Sad	
	Accuracy	Recall	Accuracy	Recall	Accuracy	Recall
Emo2vec	89.41	90.66	90.64	92.7	90.88	90.86
Generic word2vec	81.19	81.8	81.09	82.17	82.8	82.88

Table 3.3: Comparison between Emo2vec and generic word2vec approach

For the emotion sad, the highest 90.88% accuracy and 90.86% recall is achieved with codebook size 1500 and Emo2vec vector size 100.

Is Emo2vec better than generic word2vec model? Emotion specific Emo2vec training generates similar vectors for the small frames f_i, which occur in similar contexts for a specific emotion E. To evaluate the importance of such emotion specific vector generation, we compare the performance of Emo2vec against the baseline word2vec model described in section 3.2.2 under the best parameter configuration for each emotion. Our evaluation is shown in table 3.3. According to this evaluation, Emo2vec approach achieves 10.13%, 11.77% and 9.75% higher accuracy and 10.83%, 12.8% and 9.6% higher recall for emotions angry, happy and sad, respectively, compared to the baseline word2vec model.

Elimination of distorted features helpful? This section compares two cases: with and without (distorted) feature selection -i.e., considering all 231 LLD features and without the distorted ones. In section 3.1.2 we identified 48 LLD features, with less than 50% distortion over various distances to use as attributes in our robust emotion detection approach. This evaluation showed that the majority of the other LLD features we considered distort more than 100%. Including those features in our codebook generation training phase would result in assignment of various states (small frame LLD feature vectors) to wrong codebook words during testing phase, due to feature distortion in realistic settings (with reverberation, variable speaker distance, and noise).

In evaluating this issue, Table 3.4 shows that if we consider all the 231 features from section 3.1.2 in our emotion detection approach under the best parameter configuration for each emotion, the accuracy achieved for angry, happy and sad is 80.32%, 79.19%, and 80.98%, respectively. The majority of these 231 features distort significantly with noise, de-amplification and reverberation.

Emotion	Angry		Happy		Sad	
	Accuracy	Recall	Accuracy	Recall	Accuracy	Recall
With distorted feature removal	89.41	90.66	90.64	92.7	90.88	90.86
With all features	80.32	81.1	79.19	78.21	80.98	81.32

Table 3.4: Evaluation with or without distorted feature removal in Emo2vec approach

This means a state that represents a small frame LLD feature vector can deviate significantly in 231 dimensional feature space with variable speaker to microphone distances. Significant deviation of a state in feature space may result in wrong 'word' assignment from audio codebook, which may lead to wrong classification. According to this evaluation, the elimination of distorted features improves 11.3%, 14.4% and 12.2% accuracy and 11.78%, 18.5% and 11.73% recall for emotions angry, happy and sad, respectively, compared to using all 231 features. Elimination of distorted features reduces the deviation of a state in feature space, which reduces the chance of wrong 'word' assignment, hence increases accuracy.

Comparison with baselines We implemented and compared our solution to four baseline works [187, 191, 215, 170] and on our acted dataset (from section 3.4.1). A generic correlation based feature selection approach [187] performs the feature selection method to identify features with high correlation with the specific emotion class and at the same time with low correlation among themselves. This technique is not robust under realistic settings (with reverberation, variable speaker distance and noise), since many of the highly correlated features distort extensively. Hence, as shown in table 3.5, the accuracy using this approach for DER emotion detection is very low. Another baseline [191] performs 'context-aware' emotional relevant feature extraction by combining Convolutional Neural Networks (CNNs) with LSTM networks. As shown in the table, this technique of feature generation using CNN over-fits on the training data very easily and cannot adapt with feature distortion in realistic settings, hence do not perform well. [215] uses a combination of prosodic acoustic and i-vector features and uses a Recurrent Neural Network to detect speech emotion. Using this $ivector + RNN$ approach we can achieve up to 81.43%, 82.01%, and 82.32% accuracy (and 80.8%, 84.1%, and 78.43% recall) for angry, happy and sad emotion detection. According to our evaluation of section 3.1.2, the majority of the LLD features extracted in [170] distort significantly with increase of speaker to microphone distance. 39 functionals applied on those distorted features are also not robust. Since, the majority of the 6552 features distort with variable speakers distance, using this INTERSPEECH Computational Paralinguistic Challenge 13 baseline approach, we achieve low accuracy of 70.2%, 70.4% and 72.83% (68.32%, 70.88% and 70.1% recall) for angry, happy and sad emotion detection. According to table 3.5, our DER solution achieves 9.7%, 10.5%, and 10.3% higher accuracy and 12.2%, 10.2% and 15.85% higher recall compared to the best baseline solution for emotions: angry, happy and sad.

Approaches	Angry		Happy		Sad	
	Accuracy	Recall	Accuracy	Recall	Accuracy	Recall
Our solution	89.41	90.66	90.64	92.7	90.88	90.86
ivector+RNN	81.43	80.8	82.01	84.1	82.32	78.43
CNN+LSTM	74.1	76.68	73.9	73.78	76.7	73.55
Correlation based feature selection	66.1	59.8	68.9	64.6	72.99	67.54
INTERSPEECH 13	70.2	68.32	70.4	70.88	72.83	70.1

Table 3.5: Comparison with baseline

3.4.2 Evaluation: Spontaneous Family Discussions

There is no available existing mood dataset containing spontaneous speech from a realistic setting with audio recorded from multiple microphones placed at different distances. Hence, in order to evaluate our DER approach, we built our own dataset. Our in-lab protocol was developed by using examples from other in-lab family discussion protocols. Twelve families were invited to our lab and were instructed to discuss set topics related to family meals and family eating habits. Experimenters helped them initiate the discussion by providing some issues (with flexibility to discuss other related topics) that they might have wanted to change in their family and encouraged them to select a few of the topics to discuss with other members. Intuition was that, discussion about change in their existing condition would raise conflicting opinions, which in turn would encourage them to express various emotions. After initiating the discussion, experimenters left the room and let the family members have spontaneous discussions. For each of these families, we recorded 15 to 20 minutes of spontaneous discussion.

Figure 3.10 shows the lab setting, and indicates where members were seated at a round table and how far the 3 microphones were placed from the table's center: 2.25, 3 and 5.3 meters. The radius of the table was 1.5 meters, so speaker to microphone distances for Microphone 1, 2 and 3 varied from 0.7 to 3.75, 1.5 to 4.5, and 3.8 to 6.8 meters. Speakers were free to move around the table. Hence, our collected data contains speeches of moving and steady speakers from distances varied between 0.75 to 6.8 meters. Since, this project aims to recognize distant speech emotion in indoor realistic setting, variable microphone to speaker distance in range of 0.75 to 6.8 were determined considering the size of average indoor rooms [6].

In total, we collected spontaneous speech from 38 people (9 male, 29 female), with age ranging from 10 to 56 years. All of the conversations were spoken in English.

The twelve videos were coded using the Noldus Observer XT event logging software [218]. The three emotions happy, anger, and sad were coded on a 3-point scale. The emotions were labeled as one of the following three degrees: borderline emotional tone, emotional tone, or very emotional tone (e.g., borderline happy, happy, or very happy). If a coder was unsure about one of the three emotions, then were instructed to classify it as a borderline emotional tone. All the emotions were

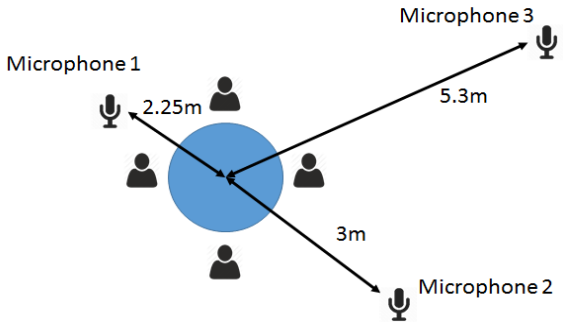


Figure 3.10: Data collection lab setting

tagged exclusively with 'start' and 'stop' times, meaning that two or more emotions were never coded at the same time for a single person. We did allow for emotions to overlap among participants (to account for participants speaking at the same time). Codes were not applied to segments where none of these three emotions were exhibited.

The two coders were research assistants with backgrounds in behavioral science. They trained on practice tapes until they achieved an inter-rater reliability rate (specifically, Cohen's kappa statistic) of 73% (0.73). This is considered moderately high [106]. The Observer XT software calculates the inter-rater reliability value automatically. After achieving a kappa statistic greater than 0.70, the two coders 'double-coded' a video every three to four discussions to ensure their inter-rater reliability rate remained above 70%. If the kappa statistic was lower than 0.70, they re-coded and re-trained until they once again achieved at least 0.70.

Training and Data Preparation

Our collected family discussions from different distance microphones contain humming sound from home appliances such as air conditioners, knocking on the table, slamming doors, squeaky hinges, dragging of chairs, etc. Hence, speech signal to noise ratio was low, specially for microphone 3 (5.3 meters distance). We have performed noise removal and filtering as shown in section 3.4.1 to reduce the steady noises. Also to adopt to home settings, we collected home environmental sounds from UrbanSound dataset [157] and use a subset of these data clips as negative samples during the training phase of our classifiers.

Our evaluation performed N fold cross validation where we trained an emotion specific classifier for each of the emotions E considering audio clips from microphone 1 (2.25 meters distance) of $N-1$ families for training and tested our trained model on audio clips from microphone 1,2, and 3 of N th family. In this evaluation we use softmax activation function in the output layer of the LSTM classifiers, which is basically the normalized exponential probability of class observations represented as neuron activations. Softmax classifiers give the probabilities for each class label. Hence, if two

Microphone Distance		2.25					3					5.3				
		Codebook size														
Emotion	Vector size	500	1000	1500	2000	2500	500	1000	1500	2000	2500	500	1000	1500	2000	2500
Angry	50	87.3	85.19	85.11	87.25	87.42	82.3	81.41	81.41	83.5	82.82	80.1	81.47	81.47	83	82.35
	100	92.48	90.1	90.1	92.52	92.52	85.19	87.28	87.28	88.9	88.86	83.98	85.11	85.19	85.9	85.9
	200	90	87.28	87.28	90.1	90.1	80	84.32	84.32	84.4	84.4	78	79.11	79.11	81.44	81.44
Sad	50	81.4	83.32	85.16	86.35	86.35	79.1	81.44	81.44	83.55	82.34	77.2	77.88	80	82.92	82.34
	100	86.36	88.72	90	90.12	90.12	85.11	85.9	85.9	88.9	87.36	82.9	84	84.31	85.11	84.31
	200	80	84.32	87.28	88.72	88.72	80	80.1	80.1	82.34	80.1	75.32	79.1	79.1	81.44	81.44
Happy	50	83.31	85.83	85.21	88.7	88.7	83.3	83.32	83.32	88.71	88.71	80	77.2	77.2	82.35	81.77
	100	88.7	90.1	90.1	94.5	94.5	88.7	88.7	88.7	92.5	92.5	84.33	86.37	86.37	87.28	86.37
	200	80	88.72	88.72	92.5	92.5	77.3	84.17	84.17	90.05	90.05	77.93	81.79	81.79	82.88	82.88

Table 3.6: Evaluation on family discussion data with various codebook and Emo2vec vector sizes

classifiers (Example: classifiers for happy and sad) provide positive output for an audio segment, we consider the classifier with higher positive output probability as the detected emotion of our DER solution.

Results

We next discuss the evaluation of our solution using the spontaneous family discussions dataset.

Change of beneficial parameters due to large variety of data? In our evaluation with the spontaneous family discussions dataset, we identify the beneficial values of t_r (threshold of rare words from section 3.2.3) and c (number of neighbour window parameter from section 3.2.3) are the same as section 3.4.1 (4, 4). The beneficial value of sub-sampling parameter t_f (from section 3.2.3) is 800. Also, our two layer LSTM classifier has 100 neurons in its first (hidden) layer.

Codebook size (section 3.2.1) influences the discriminative characteristics of our feature modeling approach. Also, our DER solution wants to identify an Emo2vec vectors size which is smaller (hence, better for computation and performance on a dataset of limited size), but not too small to loose relational and discriminative information between the words. Table 3.6 shows our evaluation with various codebook sizes and Emo2vec vector sizes on 3 different distances. According to this table beneficial codebook size for all addressed emotions and distances is 2000 and Emo2vec size is 100. Our family discussion dataset from different distance microphones contains speech from 31 individuals with variety of ages and accents. Hence, beneficial codebook and Emo2vec size is relatively larger compare to our result in section 3.4.1.

One of the significant observations from our spontaneous family discussions is, about 40% of the audio segments labeled as emotion 'happy' contains laughter. Acted emotional speech datasets (from section 3.4.1) do not contain laughter in 'happy' emotional speeches and majority of the 'false positives' for emotion: happy classifier in section 3.4.1 are 'angry' speech. Hence, presence of laughter makes classification easier for the happy emotion, which leads to higher accuracy of 94.5%, 92.5% and 87.28% (94.6%, 91.87% and 87.1% recall) for distance 2.25, 3 and 5.3 meters, respectively.

Since, our collected family discussion audio contains significant noise and reverberation, signal to

noise ratio is relatively low. Low signal to noise ratio makes it harder to distinguish between sad and other categories of speech. Hence, sad emotion detection achieved upto 90.12%, 88.9% and 85.11% accuracy (88.21%, 87.54% and 83.32% recall) for distance 2.25, 3 and 5.3 meters, respectively, which is low compared to the other two emotions (happy and angry).

Challenge: overlapping speech Approximately 15% of the voiced speech segments in our family discussion speech samples contain overlapping speech. Through our evaluation we identify that a majority of false positives from happy and angry emotion classifiers are overlapping speech. There are few studies that address the issue of identifying overlapping speech from audio, but none of them have achieved significant high accuracy. These studies [83, 173] have used spectral autocorrelation peak-valley ratio, harmonic to noise ratio, fundamental frequency, average spectral aperiodicity, MFCC, perceptual linear predictive coefficients (PLP), and spectral flatness measure as features. Using our distorted feature identification approach (section 3.1.2) we identify that MFCC, perceptual linear predictive coefficients (PLP), fundamental frequency, and spectral flatness measure distort less than 50% across various distances. Hence, we extract mel-frequency cepstral coefficients (MFCC) 1-6 in full accordance to htk-based computation, fundamental frequency computed from the cepstrum, 6 perceptual linear predictive coefficients (PLP) and spectral flatness measure from 25ms overlapping small frames. Next the 12 functionals: mean, standard deviation, kurtosis, skewness, minimum and maximum value, relative position, and range as well as two linear regression coefficients with their mean square error (MSE) are applied on a chunk of small frames (from 1 second audio segments).

Our solution trains a binary neural network classifier (with 1 hidden layer consisting of 100 neurons) with two output classes: overlapping speech and single speech, which uses the features discussed above from a 1 second audio segment as input. Softmax activation function is used in the output layer of the neural network classifiers, which gives the probabilities for each class label and the sum of probabilities of all the classes is 1. Our solution identifies a 1 second audio segment as overlapping speech and does not consider it for emotion detection only if the probability of class 'overlapping speech' for that audio segment is higher than 75%. Though our evaluation we identify that, considering this threshold enables us to detect overlapping speech with 98.1% precision and 74.8% recall. Between precision and recall trade-offs we choose to achieve the highest possible precision with a cost of lower recall. The intuition is we want to identify and exclude as much overlapping audio segments as possible, but do not (if possible) want to exclude any single speech segments from consideration for emotion classification. Table 3.7 shows the increase of DER accuracy due to our overlapping speech filtering approach. According to this table overlapping speech filtering improves about 2% accuracy for the emotion: angry across all distance. But, there is no significant improvement for emotion sad, since very few false positives for sad classifier were overlapping speech and our overlapping speech filtering approach excludes them from emotion detection consideration.

Mood	Distance (meters)	2.25		3		5.3	
		Accuracy	Recall	Accuracy	Recall	Accuracy	Recall
Angry	With filtering	94.14	95	90.9	91.1	87.55	87.53
	Without filtering	92.52	92.7	88.9	89.2	85.9	85.88
Sad	With filtering	90.1	88.3	88.9	87.54	85.9	83.87
	Without filtering	90.12	88.21	88.9	87.54	85.11	83.32
Happy	With filtering	95.26	95.77	94.11	94.2	88.76	87.34
	Without filtering	94.5	94.6	92.5	91.87	87.28	87.1

Table 3.7: Evaluation with overlapping speech filtering

Comparison with baseline As discussed in section 3.4.1 we implemented the solutions of four baseline works [187, 191, 215, 170]. Table 3.8 shows the comparison of these baseline solutions with our DER solution on the family discussion dataset (section 3.4.2). Since, the generic correlation based feature selection approach [187] and 'context-aware' emotional relevant feature extraction, by combining Convolutional Neural Networks (CNNs) with LSTM networks [191] are not robust in realistic settings (discussed in section 3.4.1), their performance is very low in our evaluation on spontaneous family discussion data. Also, majority of the 6552 features extracted in the INTERSPEECH 13 baseline approach [170], distort significantly with variable speaker to microphone distances hence, this approach achieves low accuracy in our evaluation on spontaneous family discussion data. [215] uses a combination of prosodic acoustic and i-vector features and uses Recurrent Neural Network to detect speech emotion. This $ivector + RNN$ approach achieves significantly low accuracy for our considered emotions (happy, angry and sad) compared to our evaluation in section 3.4.1, specially on 5.3 meters distance (22.5%, 24.1%, 20.1% lower accuracy and 20.8%, 30%, 22.9% lower recall on 5.3 meters distance compared to our DER approach).

3.5 Discussion

Several researches on speech analysis have used The Electromagnetic Articulography (EMA) dataset [93] in their evaluation. [101] used i-vector method and GMM classifier for speech emotion recognition and applied on EMA dataset. They did not report accuracy for particular emotions, their highest reported emotion recognition accuracy achieved was 86.765% for male speakers. Also, [12] proposed a shape-based modeling of the fundamental frequency contour for emotion detection and evaluated on EMA dataset. This study achieved 91.3%, 77.7% and 63.3% accuracy and 96%, 78% and 69.3% recall for emotions happy, angry and sad, respectively. Our Emo2vec emotion recognition approach achieves 97.2%, 96.4% and 93.8% accuracy and 98.8%, 98.2% and 96.16% recall for emotions happy, angry and sad, respectively, when applied on EMA dataset (without introducing any reverberation and de-amplification effect).

Variable speaker to microphone distances introduce noise, de-amplification of speech and room

Mood	Distance (meters)	2.25		3		5.3	
		Accuracy	Recall	Accuracy	Recall	Accuracy	Recall
Angry	Our approach	94.14	95	90.9	91.1	87.55	87.53
	ivector+RNN	79.1	78.21	77.78	77.18	70.5	67.3
	CNN+LSTM	65.76	60.3	62.89	59.19	59	56.32
	INTERSPEECH 13	66.9	67.17	63.9	64.2	56.5	59.2
	Correlation based feature selection	57.21	58.2	54.45	53.43	51.1	48.77
Sad	Our approach	90.1	88.3	88.9	87.5	85.9	83.87
	ivector+RNN	76.21	75.76	75.3	75	71.51	68.2
	CNN+LSTM	67.2	61.27	68.7	62	60	54.98
	INTERSPEECH 13	66.4	66	61.3	60.73	55.3	53.9
	Correlation based feature selection	60	58.4	56.18	55.23	53.19	53.8
Happy	Our approach	95.26	95.77	94.11	94.2	88.76	87.34
	ivector+RNN	80.6	80.61	77.78	78.43	72.4	72.3
	CNN+LSTM	70	64.9	66.8	61.84	62.67	58.61
	INTERSPEECH 13	68.47	68.4	66.91	66	59.5	56.4
	Correlation based feature selection	58.39	56.9	55.9	54.5	54.6	52.74

Table 3.8: Comparison with baseline on family discussion data

reverberation in the captured speech signal. Speech enhancement is the area of study, which aims to improve speech intelligibility and overall perceptual quality of degraded speech signal using audio signal processing techniques. Our study uses standard speech enhancement techniques such as, unvoiced audio segments removal using zero crossing rate (ZCR), noise removal using a band-pass filter and steady background noise removal using spectral noise gating algorithm to reduce noise. Hence, using these basic speech enhancement techniques we reduce background noise. However, to de-amplify speech due to increased speaker to microphone distance, it is necessary to know the distance. In general distance is not known. To the best of our knowledge there is no work which preforms blind, single channel (single microphone) speech amplification of a moving speaker across varying distances. Hence, the problem of speech de-amplification due to varying distance of a moving speaker cannot be done using speech enhancement. Further, although there are a wide variety of blind single channel (microphone) dereverberation techniques to handle room reverberation, only a few state of the art works addressed blind single channel dereverberation with moving subjects [46, 71]. Performance of these works significantly degrade with the increase of speaker to microphone distance [47]. Since, there is no good solution for either speech de-amplification or dereverberation using speech enhancement, we need a distance emotion detection (DER) solution, which is robust in emotion detection from speech with de-amplification and dereverberation.

Speech recognition is the process of capturing spoken words using a microphone and converting them into a digitally stored set of words. Speech recognition systems convert raw speech signal into a

sequence of vectors (which are a representation of the signal from small frames) which are measured throughout the duration of speech. Then, using feature modeling/engineering and a classifier these vectors are matched with phonemes. In recent years some studies have address the distant speech recognition problem [217, 151]. While speech recognition is the process of converting speech to digital data, emotion detection is aimed toward identifying the mental states of a person through speech. In this task, there is no specific definition or digital code for a vector (representation of a signal from small frames through features) to match with. Emotion detection works by analyzing the features of speech that differ between emotions. Hence, the challenges and approaches of speech recognition and emotion recognition are markedly different. Distant-emotion-recognition (DER) is an area not explored before to the best of our knowledge.

Linguistic content of the spoken utterance is also an important part of the conveyed emotion. Hence, in recent years, there has been a focus on hybrid features, i.e., the integration of acoustic and linguistic features [206, 141]. But, all of these works used clean data without addressing the challenges of variable speaker to microphone distance. Although linguistic content of the spoken utterance can help the acoustic emotion features improve the accuracy of emotion recognition, current speech recognition systems still cannot reliably recognize the entire verbal content of emotional speech. Thus, the most popular feature representation for speech recognition is acoustic features such as prosodic features (e.g., pitch-related feature, energy-related features and speech rate) and spectral features (e.g.,Mel frequency cepstral coefficients (MFCC) and cepstral features).

To evaluate automated speech to text transcription accuracy on distance speech data, in this study we asked two volunteers to translate two different spontaneous family discussions (total four family discussions, 15-20minutes each) and label the words (of their translated text) from 1 to 3, where 1 was easy to understand and 3 is extremely difficult to understand from audio clips. We performed automated speech transcription using Google Cloud API [5] (uses the most advanced deep learning neural network algorithms) on these 4 family discussions. The accuracy of automated translation on words with different difficulty levels are shown in figure 3.11. According to the figure transcription accuracy on medium and difficult words (according to human labeling) is very low, where as transcription accuracy on acted clean emotion dataset (section 3.4.1) is 96.4%.

Since, accurate transcription of distance speech signal was out of scope of this study, we focused only on distance emotion detection using acoustic signal features.

Human emotion detection from acoustic speech signal has significant potential due to its non-intrusive nature (compare to wearables) and pervasive reachability to sensors (compared to video based emotion recognitions). Hence, in recent years speech emotion detection is receiving attention with progress of advanced human-computer interaction systems [136, 139]. Also, emotion detection has paramount importance in the entertainment industry, either for the development of emotionally responsive games or toys [76] or for the development of serious games for aiding people with problems to understand social signs [18, 58]. Additionally some potential use of speech emotion detection can

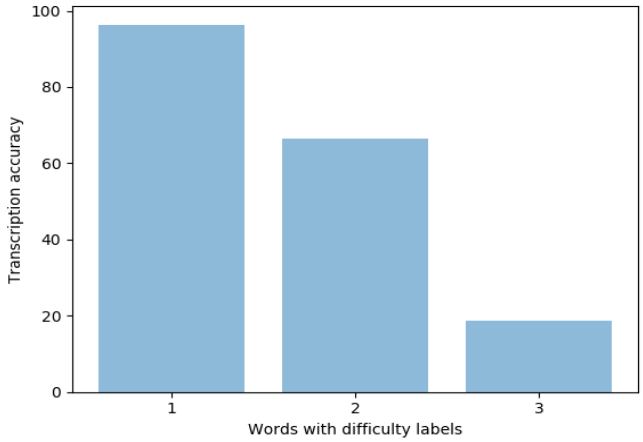


Figure 3.11: Transcription accuracy of words with different difficulty levels

be in smart homes, e-learning [34], smart vehicles [99], etc. All of these applications need distant emotion recognition and our solution has applicability for them.

3.6 Summary

Distant emotion recognition (DER) extends the application of speech emotion recognition to the very challenging situation that is determined by variable speaker to microphone distances. The performance of conventional emotion recognition systems degrades dramatically as soon as the microphone is moved away from the mouth of the speaker. This is due to a broad variety of effects such as background noise, feature distortion with distance, overlapping speech from other speakers, and reverberation.

This chapter presents a novel solution for DER, addressing the key challenges by identification and deletion of features from consideration which are significantly distorted by distance, creating a novel, called Emo2vec, feature modeling/engineering and overlapping speech filtering technique, and the use of an LSTM classifier to capture the temporal dynamics of speech states found in emotions.

The end result (Automated DER) [158] has usability in non-intrusive monitoring of people with a variety of physical and mental health problems, as well as for human computer interaction systems, smart homes, the entertainment industry, and many other domains.

Chapter 4

Mental Disorders: Weakly Labeled Audio Data

Conventional supervised learning audio event detection classifiers need annotated data, where the segments of audio containing the desired vocal event are clearly indicated. We refer to such data as strongly labeled data. But strong labeling of mental disorders (e.g., social anxiety disorder, depression, etc.) in speech audio clips is impractical, because, it is not possible to identify with high confidence which regions of a conversation or long speech indicate an individual's mental disorder.

A solution is to collect long speech audio samples from individuals already diagnosed with or high in symptoms of specific mental disorders from situations that may heighten expression of the symptoms of respective disorders. This type of data is considered "weakly labeled," meaning that although they provide information about the presence or absence of disorder symptoms, they do not provide additional details such as the precise times in the recording that indicate the disorder, or the duration of those identifying regions.

This chapter discusses the task of identifying speakers high in symptoms of two distinct mental disorders: social anxiety disorder and depression from weakly labeled audio data. We expected that recordings from individuals high in symptoms would have regions indicative of those symptoms (i.e., regions present for persons high, but not for persons low, in symptoms).

This approach falls under the general rubric of multiple instance learning (MIL). MIL is a weakly supervised learning approach in which labels for individual instances are unknown; instead, labels are available for a collection of instances, usually called "bag." A positive bag has at least one positive instance (indicating high symptoms) and may contain negative instances (label noise), whereas a negative bag contains negative instances only. In this study, we break weakly labeled audio clips into several small, contiguous segments, where the segments are the instances and the audio clip is the bag.

To our knowledge, no previous research has identified individuals high in symptoms of a mental disorder from weakly labeled audio data. The contributions of this study are:

- We present a novel weakly supervised learning framework for detecting individuals high in symptoms of two mental disorders (social anxiety and depression) from weakly labeled audio data, adding a practical complement to health-care providers' assessment modalities.

- We propose a novel feature modeling/engineering technique named NN2Vec (section 4.1.3) to generate low-dimensional, continuous, and meaningful representation of speech from long weakly labeled audio data. All existing techniques (e.g., I-vector, audio words, Emo2Vec) are designed for strongly labeled data; hence, they fail to meaningfully represent speech due to the significant label noise in weakly labeled audio. NN2Vec identifies and exploits the inherent relationship between audio states and targeted vocal events. Identifying individuals high in social anxiety and depression symptoms using NN2Vec achieved on average F-1 scores 17% and 13% higher, respectively, than those of the other techniques (sections 4.4.1 and 4.5).

- MIL adaptation performs significantly better than supervised learning classifiers (where the individual instance labels are ambiguous), which fall short of generating an optimal solution due to label noise in weakly labeled data [28]. Studies have shown that emotion or mental disorder can be perceived by the temporal dynamics across speech states [158, 215, 91, 100]. To generate a sequential deep neural network solution that comprehends the temporal properties in speech while also being adaptive to noise in weakly labeled long audio data, we developed a novel MIL adaptation of bidirectional long short term memory classifier, named BLSTM-MIL (section 4.2.1).

- Because no existing dataset contained spontaneous speech labeled with speakers high in social anxiety, we built a dataset consisting of 3-minute samples of weakly labeled spontaneous speech from 101 participants. Our approach achieves an F-1 score of 90.1% in detecting speakers high in social anxiety symptoms. Additionally, our approach achieves an F-1 score of 93.4% in detecting anxious versus calm states based on participants' self-reported levels of peak emotion during the speech.

- We analyzed data from a publicly available Distress Analysis Interview Corpus (DAIC-WOZ) database [193] that contains weak labels of participants' mental disorder (depressed vs. non-depressed) on 10-15 minute interviews. Our approach achieves an F-1 score of 85.44% in detecting speakers with depression, which is 33% higher than that of the best state-of-the-art work evaluated on this dataset (section 4.5).

4.1 Feature Modeling

In the following sections, we discuss extracted LLDs from raw audio signal (section 4.1.1) and use an audio-codebook approach to map the audio signal to audio words (section 4.1.2) and our new NN2Vec feature modeling approach (section 4.1.3) to reflect the audio information for MIL algorithms. Our novel MIL solution uses NN2Vec with a new BLSTM-MIL (section 4.2.1) classifier to detect vocal events from weakly labeled data.

4.1.1 Audio Features

Our approach segments the audio clips into overlapping windows and extracts a feature set from each window. Extracted feature sets represent the inherent state of audio from that window. Based on the previous studies on audio features associated with human vocal event detection (section **??**), we considered the LLDs shown in the left column of table 4.1, as well as their delta and delta-delta coefficients. Each window is segmented into overlapping 25-ms frames with 10-ms overlap, from which LLDs are extracted. Next, the 8 functionals shown in the right column of table 4.1 are applied to extract the audio window representation. In total, 272 features are extracted from each of the overlapping windows. We evaluated window size from 500 ms to 10 seconds.

Features	Functionals
Zero crossing rate & Δ (2-dim) Energy & Δ (2-dim) Spectral centroid & Δ (2-dim) Pitch & Δ (2-dim) MFCC & Δ (26-dim)	Min, Max, std, var, mean, median, skew, and kurtosis

Table 4.1: Low-level descriptive features and high-level functionals; *std*: standard deviation; *var*: variance; *dim*: dimension

4.1.2 From Audio to Words

We use the audio-codebook model [129, 150] to represent the audio signal in a window with audio words. The audio words are not words in the typical, semantic meaning of words, but rather fragments of the audio signal represented by features. We need robust features to represent the audio state in a window. Inspired by [85], we use a GMM-based clustering method to generate the audio codebook from the functional representations mentioned in section 4.1.1.

To generate the codebook, a GMM-based model is trained on randomly sampled data from the training set. The resulting clusters form the codebook audio words. Once the codebook is generated, acoustic HLDs within a certain range of the audio signal are assigned to the closest audio words (GMM cluster centers) in the codebook.

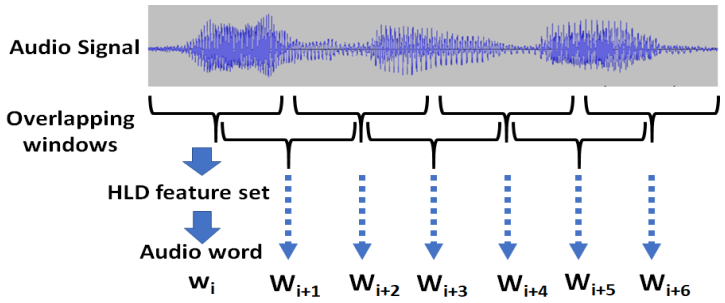


Figure 4.1: Conversion of audio signal to sequence of audio words using audio-codebook method

The discriminating power of an audio-codebook model is governed by the codebook size. The codebook size is determined by the number of clusters C generated by the GMM. In general, larger codebooks are thought to be more discriminating, whereas smaller codebooks should generalize better, especially when HLDs extracted from frames can be distorted with distance, environmental noise, and reverberation, as smaller codebooks are more robust against incorrect assignments. However, a codebook that is too small is too generic, and, hence, unable to capture the change in speech states in various small frames. Hence, through the audio-codebook approach, audio clips are converted to a sequence of audio words.

4.1.3 NN2Vec Approach

Human emotion or mental states are represented by a sequence of audio states [158, 100, 95], which are represented by audio words. Our assumption is that regions (subsequences of audio words) indicative of targeted vocal events (high symptom/disorder classification) are common (occur with high probability) across positive audio clips, and not present or rarely present (occur with low probability) in negative audio clips. MIL requires that the feature modeling learn the inherent relation between audio states and vocal events (positive class) and that the generated feature representations indicate the positive class. Conventional feature modeling (audio word, I-vector, etc.) techniques cannot learn this relation effectively from weakly labeled long audio clips (section 4.4.1).

To identify and exploit the inherent relationship between audio states and vocal events from weakly labeled data, we developed a neural network-to-vector conversion (NN2Vec) approach that generates an N dimensional dense vector representation for each of the audio words. The contributions of NN2Vec are:

- **Representational efficiency:** Audio word representation relies on the notion of one hot encoded vector, where an audio word is represented by a sparse vector with a dimension equal to the size of the vocabulary with a 1 at the index that stands for the word and 0s everywhere else. Hence, the feature representation dimension is significantly high, which is

difficult for a classifier to optimize using limited weakly labeled data. NN2vec is a shallow neural network model that generates a fixed-length dense vector for each of the audio words. This means that the model learns to map each discrete audio word representation (0 through the number of words in the vocabulary) into a low-dimensional continuous vector space from their distributional properties observed in training. This is done by a relatively straightforward optimization that starts with a more or less random assignment and then progressively reduces the overall error with a gradient descent method. We evaluated with codebook sizes V from 500 to 5000 and found that the best NN2Vec dimension N is between 20 and 50, based on V. Hence, compared to high-dimensional sparse audio word features, NN2Vec features represent audio states with significantly low-dimensional distributed representation.

- **Mapping efficiency:** An interesting property of NN2vec vectors is that they not only map the states of audio (audio words) in a smaller space, but also encode the syntactic relationships between audio states. NN2Vec vectors are similar for audio states with similar probability of occurring in positive audio clips. Neural networks typically respond in a similar manner to similar inputs. Generated distributed representations are designed to take advantage of this; audio states that should result in similar responses are represented by similar NN2Vec vectors, and audio states that should result in different responses are represented by quite different NN2Vec vectors. Hence, identification of sequences of states indicative of a mental disorder should be easier for a weakly supervised classifier.

- **Continuity:** Representing states in continuous vector space allows powerful gradient-based learning techniques such as backpropagation to be applied effectively. Previous studies [135, 140] have shown that distributed representation of input features improves classification performance compared to discrete representation.

NN2Vec Vector Generation

This subsection describes our NN2Vec feature generation approach, where NN2Vec vector representations of audio states are learned by a fully connected neural network. Later sections discuss how our NN2Vec neural network model learns similar vector representations for audio words with similar probability of occurring in positive audio clips. Our training set contains (B_i,Y_i) audio clip-label pairs, where the ith audio clip is denoted as B_i and its corresponding label $Y_i \in \{1, 0\}$. We segment these clips into overlapping windows, s_{ij} (jth window in audio clip B_i) and assign an audio word-label pair (w_{ij}, y_{ij}) to each of them. Here, w_{ij} is the audio word extracted through the audio-codebook approach (section 4.1.2) from window s_{ij} and $y_{ij} = Y_i$, label of the respective audio clip. Considering that a codebook size (section 4.1.2) is V, these audio words w_{ij} are converted to a V dimensional one-hot encoded vector X_{ij}. Suppose audio word w_{ij} is the lth audio word in the codebook. Then its one-hot vector representation would be: $X_{ij} = [x_{ijk}]$, where $k = 1 \dots V$

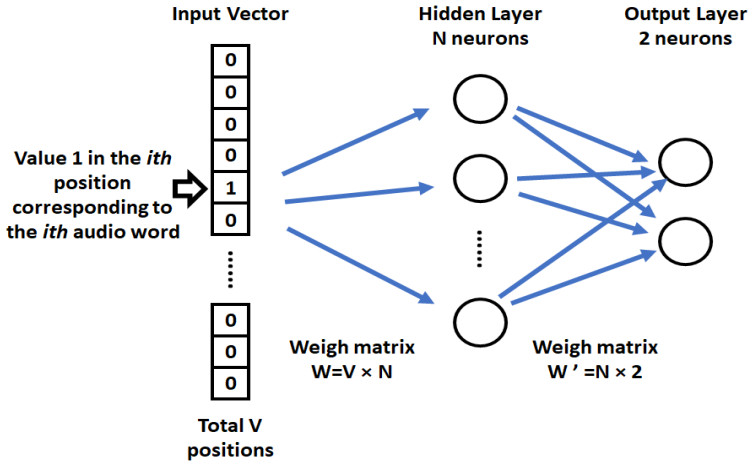


Figure 4.2: NN2Vec fully connected neural network

and $x_{ijk} = 1$, only if $k = l$ and $x_{ijk} = 0$ otherwise. These one-hot vector-segment level label pairs (X_{ij}, y_{ij}) are our training input set for the NN2Vec vector generation model.

Figure 4.2 shows our NN2Vec fully connected neural network. Here, the input layer is V dimensional, corresponding to one-hot vectors, and the output layer is a 2-dimensional softmax layer. If the hidden layer has N neurons, generated NN2Vec vectors would be N dimensional. We train the network with (X_{ij}, y_{ij}) pairs from the training set. The weights between the input layer and the output layer of the NN2Vec network can be represented by a $V \times N$ matrix W. Each row of W is the N-dimension vector. After training each row r of W is our NN2Vec vector representation of the rth audio word in the codebook (section 4.1.2). Through this approach, if two audio words occur with similar frequency (hence, similar probability) in positive class examples, their corresponding rows in W, hence generated NN2Vec vectors would be similar.

Learning Vector Representations Through NN2Vec Model

This subsection discusses the approach through which the NN2Vec model learns similar vector representations for the audio words that occur with similar probability in a targeted audio event (positive class examples). Figure 4.3 shows the simplified form of the NN2Vec network model. Suppose the codebook size (section 4.1.2) is V and the hidden layer size is N, which means the generated NN2Vec vector size would be N. All the layers are fully connected layers. The input is a one-hot encoded vector, which means that for a given input audio word, only one out of V units, $\{x_1, x_2, \ldots, x_V\}$, will be 1, and the rest will be 0.

The weights between the input layer and the hidden layer can be represented by a $V \times N$ matrix W. Each row of W is the N-dimensional vector representation v_w of the associated audio word of

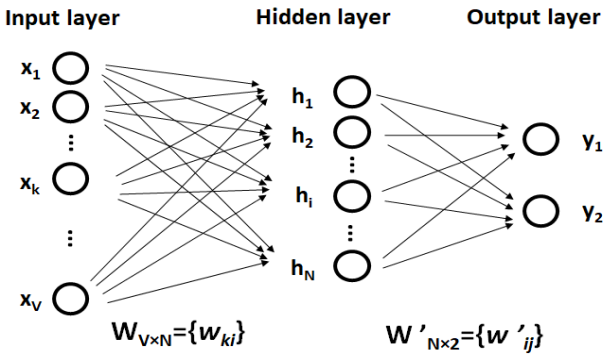


Figure 4.3: NN2Vec model with binary output

the input layer. Given an audio word w_k, $x_k = 1$ and $x_{k'} = 0$ for $k^{'} \neq k$, and:

$$h = W^T x = W^T_{(k,.)} := v^T_{w_k} \tag{4.1}$$

, which is the k row of W to h. v_{w_k} is the vector representation of the input audio word w_k .

There is a different weight matrix $W^{'} = \{w^{'}_{ij}\}$ from the hidden layer to the output layer, which is a $N \times 2$ matrix. Using these weights, we calculate the score u_j for each class.

$$u_j = {v^{'}_{c_j}}^T h \tag{4.2}$$

Here, $v^{'}_{c_j}$ is the jth column of matrix $W^{'}$. The NN2Vec architecture uses softmax, a log-linear classification model to calculate the posterior probability, which is a multinomial distribution.

$$p(c_j|w_k) = y_j = \frac{exp(u_j)}{\sum^2_{j'=1} exp(u_{j'})} = \frac{exp({v^{'}_{c_j}}^T v_{w_k})}{\sum^2_{j'=1} exp({v^{'}_{c_{j'}}}^T v_{w_k})} \tag{4.3}$$

Here, y_j is the output of the j unit in the output layer (in total 2 classes), v_{w_k} is the vector representation of the input audio word w_k and $v^{'}_{c_j}$ is the representation of class c_j.

Weight updates of this network are performed by backpropagation [67] where the training objective is to maximize the conditional probability of observing the actual output class c_O, given the input audio word w_I (as shown in equation 4.3) with regard to the weights. Here, O denotes output class and I denotes the input audio word index. The loss function is $E = -log\ p(c_O|w_I)$, which we want to minimize, and the network prediction error of jth-output unit $e_j = \frac{\partial E}{\partial u_j} = y_j - t_j$ is the derivative of E with regard to the jth output layer unit's network input u_j. Here t_j will only be 1 when the j-th unit is the actual output class, otherwise $t_j = 0$.

Using stochastic gradient descent, the weight-updating equation for hidden layer to output

weights ($W^{'}$) is:

$$w_{ij}^{'\,new} = w_{ij}^{'\,old} - \eta.e_j.h_i \tag{4.4}$$

or

$$v_{c_j}^{'\,new} = v_{c_j}^{'\,old} - \eta.e_j.h \text{ for } j = 1, 2 \tag{4.5}$$

,where $\eta > 0$ is the learning rate, $h = v_{w_I}^T$, and $v_{c_j}^{'}$ is the vector representation of class j. Hence, if $y_j > t_j$, then a portion of the hidden vector h (i.e., v_{w_I}) is subtracted from $v_{c_j}^{'}$, making $v_{c_j}^{'}$ further away from v_{w_I}; if $y_j < t_j$ (only when $t_j = 1$; i.e., $c_j = c_O$), then a portion of the hidden vector h (i.e., v_{w_I}) is added to $v_{c_O}^{'}$ (here, $j = O$), making $v_{c_O}^{'}$ closer to v_{w_I}.

Moreover, the weight-updating equation for input layer to the hidden layer weights (W) is:

$$v_{w_I}^{\,new} = v_{w_I}^{\,old} - \eta.EH^T \tag{4.6}$$

Here, v_{w_I} is a row of W, the input vector representation of the audio word I, and is the only row of W whose derivative on the loss function ($\frac{\partial E}{\partial W}$) is non-zero, given that inputs of the NN2Vec model are one hot encoded vectors. Hence, all the other rows of W will remain unchanged after this iteration, because their derivatives are zero. Also, $EH_i = \sum_{j=1}^{2} e_j.w_{ij}^{'}$, where $w_{ij}^{'}$ is the i-th hidden layer unit to j-th output layer unit weight. Hence, vector EH is the sum of output vectors of all classes (two in our case) weighted by their prediction error e_j. Therefore, equation 4.6 essentially adds a portion of two output vectors to the input vector of the input audio word.

The movement of the input vector of w_I is determined by the prediction error; the larger the prediction error, the more significant effects an output vector of a class will exert on the movement on the input vector of audio word. If, in the output layer, the probability of a class c_j being the output class c_O is overestimated ($y_j > t_j$), then the input vector of the audio word w_I will tend to move farther away from the output vector representation of class c_j; conversely, if the probability of a class c_j being the output class c_O is underestimated ($y_j < t_j$), then the input vector of the audio word w_I will tend to move closer to the output vector representation of class c_j. If the probability of class c_j is fairly accurately predicted, there will be very small movement on the input vector w_I.

As the model parameters update iteratively in each epoch by going through audio word to target class pairs generated from training data, the effects on the vectors accumulate. The output vector representation of a class c is moved back and forth by the input vectors of audio words w which occur in that class (c) in the training data (equation 4.5), as if there were physical strings between the vector of c and the vectors of audio words. Similarly, an input vector of an audio word w can also be considered as being moved by two output vectors (equation 4.6). The equilibrium length of each imaginary string is related to the strength of co-occurrence between the associated audio word and class pair.

Given our proposed NN2Vec is a binary softmax classification model, for an audio word w_k, the $p(c_1|w_k) = 1 - p(c_0|w_k)$. Here, c_1 is the positive and c_0 is the negative class. Hence, we can consider

that during training an input vector of an audio word w will be moved by an output vector of positive class c_1. After many iterations, the relative positions of the input (for audio words) and output (for class) vectors will eventually stabilize. As stated before, the relative position or similarity of these input-output vector pairs depends on the frequency of these pairs in training data, which means the probability of an audio word w_k occurs in class c_1 (positive event), $p(c_1|w_k)$.

Now, consider two audio words w_p and w_q that occur with similar probability in class c_1. Their relative vector representation will be similar compared to an output vector of class c_1. That means the vector representations of w_p and w_q will be similar. Hence, all the audio words that occur with similar probability to occur in positive class audio clips (in training set), would have similar NN2Vec vector representation through our proposed NN2Vec model approach.

4.2 Multiple Instance Learning Solution

Our tasks are binary classification tasks where labels are either -1 or 1. MIL is a kind of weakly-supervised learning. Each sample is in the form of labeled bags, composed of a wide diversity of instances associated with input features. Labels are attached to the bags, rather than to the individual instances within them. A positive bag is one that has at least one positive instance (an instance from the target class to be classified). A negative bag contains negative instances only. A negative bag is thus pure, whereas a positive bag is impure. This assumption generates an asymmetry from a learning perspective as all instances in a negative bag can be uniquely assigned a negative label, which cannot be done for a positive bag (which may contain both positive and negative instances).

We represent the bag-label pairs as (B_i, Y_i). Here, the ith bag is denoted as B_i, of size l_i, and the jth instance in the bag as x_{ij} where $j \in 1 \dots l_i$. The label for bag i is $Y_i \in \{-1, 1\}$, and the label for instance x_{ij} is y_{ij}. The label y_{ij} for instances in bag B_i can be stated as:

$$Y_i = -1 \implies y_{ij} = -1 \; \forall \; x_{ij} \in B_i \tag{4.7}$$

$$Y_i = 1 \implies y_{ij} = 1 \text{ for at least one } x_{ij} \in B_i \tag{4.8}$$

This relation between Y_i and y_{ij} is: $Y_i = \max_j\{y_{ij}\}$

Hence, the MIL problem is to learn a classification model so that given a new bag B_t it can predict the label Y_i.

To classify (binary) our weakly labeled data, we break the audio clips into several small contiguous segments. Considering reasonably-sized segments, it is safe to assume that if an audio is labeled as a positive class (anxious mental state above 0 based on self-reported peak anxiety during the speech or high symptom/disorder classification based on screening measure), then at least one of the segments is a positive example, containing a region or pattern indicative of the positive class. On

the contrary, if an audio is labeled as a negative class, none of the segments will contain a region or pattern indicative of positive class (i.e., hence all segments are negative examples). Hence, according to the MIL definition, the audio clips can be treated as bags B_i and the segments as instances x_{ij} of the corresponding bag. From the arguments just stated, if the weak information identifies the presence of a positive class in an audio segment, then the label for the corresponding bag is +1. Otherwise, it is -1.

A variety of MIL algorithms have been proposed in the literature. This study considers two MIL algorithms as baselines and presents one novel BLSTM-MIL algorithm for MIL. The first baseline algorithm (miSVM) [11] is based on Support Vector Machine (SVM). The standard SVM algorithm is modified to work in the MIL domain. Although a few other formulations of SVM for MIL domain have been proposed [43], the miSVM is the first SVM formulations for MIL and performs well on a variety of MIL tasks. The second baseline algorithm (DNN-MIL) [207] is a deep neural network modified for MIL domain. These MIL classifiers [207, 11, 43] extract a feature vector from each of the segments that are considered to be a representation of an instance. Hence, these classifiers [207, 11, 43] fail to capture the temporal dynamics of speech states, which is indicative of vocal events [158, 100, 95]. In the following section, we discuss our novel MIL method (section 4.2.1) based on a long short-term memory classifier.

4.2.1 BLSTM-MIL

Comprehension of temporal dynamics of states throughout a speech segment (which contains the region or pattern indicative of high symptom/disorder classification) requires long-term dependency. The BLSTM classifier takes sequential inputs where the hidden state of one time step is computed by combining the current input with the hidden state of the previous time steps. They can learn from current time step data as well as use relevant knowledge from the past to predict outcomes. Hence, we present a BLSTM-MIL classifier that uses the temporal information of speech states within an audio segment (which represents an instance) to learn the instance label.

Our BLSTM-MIL classifier is shown in figure 4.4. An audio clip is segmented into overlapping windows, and feature sets (i.e., NN2Vec vectors) are extracted from each of these windows. In this approach, the feature sets extracted from the windows represent the state of audio from the respective windows. Feature sets from m consecutive windows comprise an instance x_{ij} of bag B_i. For example, the feature set of the kth window of jth instance x_{ij}, from bag B_i is denoted by f_{ijk}, where $k = 1 \dots m$. Hence, each instance (MIL representation of a segment) contains representations of audio states (feature sets) as well as their changes (sequence of feature sets) throughout time. In figure 4.4, $m = 4$ with overlapping size 2, which means 4 consecutive feature sets (representation of audio states of 4 consecutive windows) comprise an instance x_{ij}.

A sequence of feature sets f_{ijk}, with $k = 1 \dots m$, for each instance x_{ij} in a bag B_i is first fed into the 2-layer BLSTM network with an activation function (in this study we use sigmoid activation

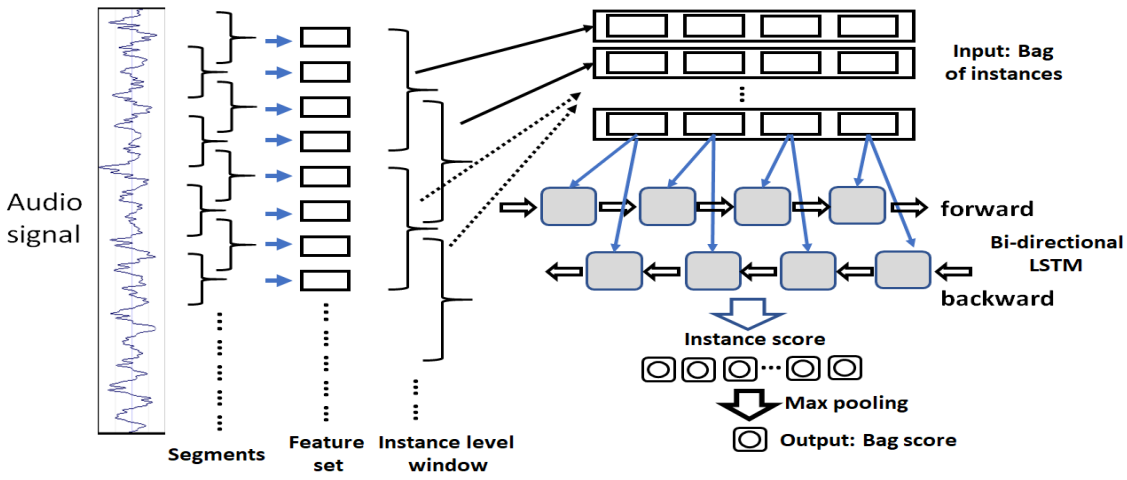


Figure 4.4: Bidirectional LSTM multiple instance learning classifier (BLSTM-MIL)

[177]). Through this architecture, we can efficiently make use of past features (via forward states) and future features (via backward states) for a specific time window. Hence, we can capture the forward and backward temporal progression of audio states within a time window (which represent the instance x_{ij}). The forward and backward passes over the unfolded network over time are carried out in a similar way to regular network forward and backward passes, except that we need to unfold the hidden states for all time steps. We do forward and backward propagation for entire audio clips, and we only need to reset the hidden states to 0 at the beginning of each audio clip.

The last layer of the network is a MIL Max Pooling Layer. The MIL Max Pooling layer takes the instance level probabilities o_{ij} for instances x_{ij} of a bag B_i as input and predicts bag label denoted as Y_i^o, according to the following equation:

$$Y_i^o = 1 \text{ if } \max_j o_{ij} \geq \tau, \text{ or } Y_i^o = -1 \text{ otherwise} \tag{4.9}$$

According to this equation, if at least one of the instance level probabilities is greater than the threshold τ, the predicted bag level would be 1, and -1 otherwise.

The MIL adaptation of BLSTM is trained using backpropagation using the gradients of divergence shown in equation 4.10 & 4.11.

$$E_i = \frac{1}{2}(\max_{1 \leq j \leq n_i} (o_{ij}) - d_i)^2 \tag{4.10}$$

$$E = \sum_{i=1}^{N} E_i \tag{4.11}$$

Here, d_i is the desired output in response to the set of instances from bag B_i. d_i is set to Y_i, the

label assigned to B_i.

In BLSTM-MIL training, all the instances x_{ij} of one bag B_i are considered as a batch (input), and single gradient update (updating network parameters) is performed over one batch of samples. During training, once all instances in a bag have been fed-forward through the network, the weight update for the bag is done with respect to the instance in the bag for which the output was maximum. The process is continued until the overall divergence E falls below a desired tolerance. Because all the instances of one bag are inputted as a batch (during training and testing) and the number of instances in a batch size can vary, BLSTM-MIL is adaptable to variable size audio clips. Conventional neural networks (e.g., DNN, CNN) are constrained on fixed size input.

In weakly labeled data, all the instances of a negative bag (training sample) are negative. But in positive training samples, only a small portion of the instances are positive and the rest are negative (noisy instances). Training supervised learning neural network classifiers (e.g., DNN, CNN, BLSTM) considers labels of all the instances of a positive training bag as positive. Due to the significant amount of label noise in positive training samples, supervised learning neural network approaches fail to achieve an optimal solution.

By contrast, in BLSTM-MIL training (equation 4.10), if at least one instance of a positive training bag is perfectly predicted as positive, the error E_i on the concerned bag is zero and the weights of the network will not be updated. Therefore, the BLSTM-MIL network training avoids weight updates due to noisy instances in positive training samples. Additionally, if all the instances of a negative bag are perfectly predicted as negative, then only the error E_i (equation 4.10) on the concerned bag is zero and the weights of the network are not updated. Otherwise, the weights are updated according to the error on the instance whose corresponding actual output is maximal among all the instances in the bag.

The ideal output of the network in response to any negative instance is 0, whereas for a positive instance it is 1. For negative bags, equation 4.10 characterizes the worst-case divergence of all instances in the bag from this ideal output. Minimizing this ensures that the response of the network to all instances from the bag is forced towards 0. In the ideal case, the system will output 0 in response to all inputs in the bag, and the divergence E_i will become 0.

For positive bags, equation 4.10 computes the best-case divergence of the instances of the bag from the ideal output of 1. Minimizing this ensures that the response of the network to at least one of the instances from the bag is forced towards 1. In the ideal case, one or more of the inputs in the bag will produce an output of 1, and the divergence E_i becomes 0.

Hence, using equations 4.10 and 4.9 during training and testing, the MIL adaptation of BLSTM treats negative training samples as supervised learning approaches do, given that negative samples do not contain noisy labels, but effectively avoids weight updates due to noisy labels in positive samples.

4.3 Datasets

This section describes the weakly labeled audio datasets for our evaluation of high social anxiety and depression. We discuss the evaluations themselves in sections 4.4 and 4.5, respectively.

4.3.1 Social Anxiety

Because no previous dataset contained sponteneous speech labeled with speakers high in social anxiety, we built our own dataset from a laboratory-based study of a university student sample. The study was approved by the University of Virginia Institutional Review Board (IRB protocol 2013-0262-00) and conducted under the supervision of a licensed clinical psychologist and researcher with expertise in anxiety disorders. Because the collected audio data contains personal content and potentially identifying characteristics, this dataset cannot be shared with outside researchers given the need to protect confidentiality.

Participants

A total of 101 participants ranging from 17 to 18 years of age (M = 19.24, SD = 1.84) completed the study in exchange for course credit or payment. Participants reported their races as 73.8% White, 13.4% Asian, 6.4% Black, 3.7% multiple, and 2.1% other (0.5% declined to answer) and their ethnicities as 90.9% Non-Hispanic/Latino and 7.0% Hispanic/Latino (2.1% declined to answer).

Participants were selected based on a screening survey at the start of the semester that included the Social Interaction Anxiety Scale (SIAS) and an item from the Social Phobia Scale (SPS) assessing anxiety about public speaking ("I get tense when I speak in front of other people" [105]). We included both measures because the speech task required participants to have anxiety about public speaking (it is possible to have social anxiety symptoms from fearing other social situations, such as dating, without fearing public speaking). Participants with SIAS scores less than or equal to one-quarter of a standard deviation (17 or less) below the mean of a previous undergraduate sample (M = 19.0, SD = 10.1; [105]) and who rated the public-speaking item as 0 (*not at all*), 1 (*slightly*), or 2 (*moderately*) were invited to join the Low Social Anxiety (Low SA) group; 56 enrolled. Those scoring greater than or equal to one standard deviation (30 or greater) above the SIAS mean and who rated the public-speaking item as 3 (*very*) or 4 (*extremely*) were invited to join the High Social Anxiety (High SA) group; 45 enrolled. This screening method or directly analogous ones were used in previous studies [32, 33]. The mean SIAS score for the High SA group (45.9, SD = 10.6) was close to the mean reported for a socially phobic sample (49.0, SD = 15.6; [68]), suggesting a strong analog sample.

Speech Task

Participants were told researchers are interested in learning how people perceive and predict their own speaking abilities, utilizing a set of guidelines for effective speaking that the researchers

were developing. Social anxiety was not mentioned and participants did not know why they were invited to participate (until full debriefing after the study). As part of a larger study, participants were instructed to give an approximately 3-minute speech to the best of their ability on things they liked and disliked about college or their hometown in front of a large cassette video camera (to make the recording salient) with a web camera used for actual recording mounted on top. They were told the speech was being videotaped so that another researcher in an adjacent room would be able to watch and evaluate their performance. Videotaping was done to make the cover story as believable as possible and to heighten the participants' anxiety (following [41]). The present study evaluates classification approaches only on these speeches (M length = 3 minutes). Participants were offered 1 minute to prepare their speech. If a participant paused for a significant period of time during the speech, the experimenter encouraged (but did not force) the person to continue talking for the full time.

4.3.2 Depression

Distress Analysis Interview Corpus - Wizard of Oz (DAIC-WOZ) is a public dataset made available as part of the 2016 Audio-Visual Emotion Challenge and Workshop (AVEC 2016) [193]. It contains audio (with transcription) and video recordings of interviews from 189 participants in English with an animated virtual interviewer operated via a Wizard-of-Oz paradigm [61]. Each participant was assigned a score on the Patient Health Questionnaire-8, a self-report of depressive symptoms [84]. As part of the AVEC challenge [193], 142 participants were assigned to one of two classes: depressed (42) or non-depressed (100); the mean scores were 15.9 and 2.75, respectively. The present study uses only the audio data (M length = 12 minutes, range= 10-25 minutes).

4.4 Evaluation: Social anxiety

This section describes our evaluations of NN2Vec and BLSTM-MIL on social anxiety data (section 4.3.1) for detecting speakers high in social anxiety symptoms (section 4.4.1).

All our evaluations were performed using leave-one-speaker-out cross-validation. To avoid overfitting, we randomly selected 30% of the audio clips for training the audio word codebook (section 4.1.2) and 30% of the audio clips to train the NN2Vec model to generate vector representations (section 4.1.3) in each evaluation.

4.4.1 Social Anxiety Group

In this section we discuss the efficiency and applicability of our solution by investigating some key questions.

What are beneficial parameter configurations?

There are a number of parameters in our solution. Segment (which represents an instance in MIL) size is one important parameter. If the segment size is too small, it may contain only a fraction of the region indicating the positive class. If segment size is too large, the region indicating the positive class can be only a small fraction of the audio segment, and feature representation and the MIL classifier may fail to comprehend the indicative patterns.

A grid search over window size (section 4.1.1) from 500 ms to 10 seconds revealed that 1-second window size performs better on average.

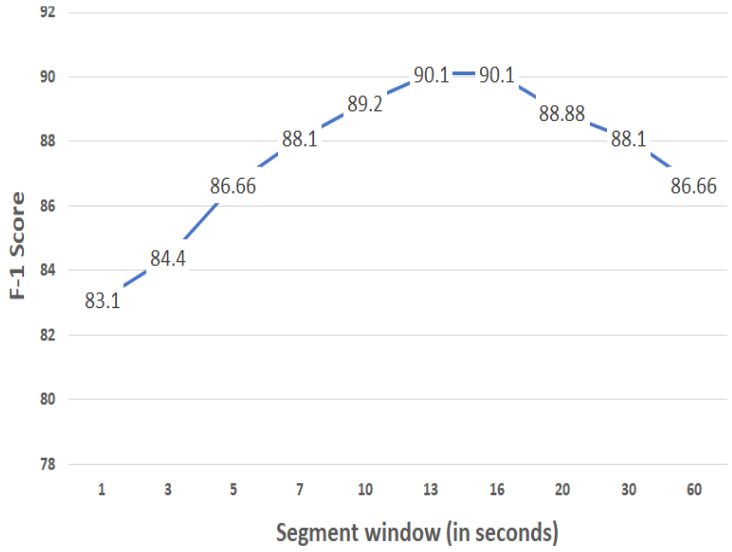


Figure 4.5: Evaluation for social anxiety with variable segment size

Figure 4.5 shows our evaluation with various MIL instance segment sizes. As shown in figure 4.5, the F-1 score increases from 83.1% to 90.1% as instance segment size increases from 1 second to 13 seconds under the best parameter configuration. From 13 to 20 seconds the F-1 score is similar and then decreases as segment size increase further. Thus, the optimal segment instance size is 13 seconds.

Our BLSTM-MIL classifier has two layers with [100, 100] nodes, a 20% dropout rate, and two fully connected dense layers [20, 1] with sigmoid activation function. In the BLSTM-MIL approach, an instance segment is comprised of a sequence of NN2Vec vectors, which means that instance segments are sequences of audio states. BLSTM can learn from the current audio state as well as use relevant knowledge from the past to predict outcomes. If a region indicating the positive class (high symptom/disorder classification) is smaller than the segment window, BLSTM can still pass the knowledge through hidden states from one time step to another. Hence, for instance segment

sizes 10 seconds to 20 seconds, our F-1 score is similar (88.88% to 90.1%). But as the segment size increases, the regions indicating the positive class decrease more and more relative to the instance segment; hence, BLSTM-MIL performance starts to decrease.

Audio codebook size	F-1 Score	Accuracy
500	77.2	78.9
1000	82.8	84.1
2000	87.9	89.1
2500	88.17	89.1
3000	89.13	90.1
3500	90.1	91
4000	90.1	91
4500	89.1	90
5000	88.17	89.1

Table 4.2: Evaluation for social anxiety with variable audio codebook size

Next, we evaluated our solution with various codebook sizes (table 4.2), setting segment instance size to 13 seconds and window size to 1 second. When we evaluated NN2Vec vector (section 4.1.3) dimension from 10 to 100, we found that a vector dimension between 30 to 50 performs better on average for all codebook sizes. Thus, we extracted 30 dimensional NN2Vec vectors in our evaluation of codebook size. According to this evaluation, with an audio codebook size of 3500 we achieve the highest performance of an F-1 score of 90.1% and 91% accuracy.

Is NN2Vec Better?

We evaluate our BLSTM-MIL implementation (section 4.2.1) using NN2Vec feature representation against four baselines: audio words, I-vector, Emo2vec, and raw audio features (section 4.1.1). Table 4.3 shows the results.

Feature	F-1 Score	Accuracy
NN2Vec	90.1	90
Emo2vec	72.3	77.22
I-vector	74.7	79
Audio word	55.55	68
Raw features	56.82	62.4

Table 4.3: Evaluation for social anxiety with NN2Vec and various feature representations

The I-vector system is a technique [82] to map the high-dimensional GMM supervector space (generated from concatenating all the mean values of GMM) to low-dimensional space called total variability space. The basic idea of using an I-vector in human event detection [96, 208] is to represent each instance (window) using concatenated I-vector feature vectors extracted based on event-specific (e.g., emotion) GMM super vectors, and then to use these in the classifiers. Hence, the first step is

Algorithm	F-1 Score	Accuracy
BLSTM-MIL	90.1	90
DNN-MIL	85	88.11
mi-SVM	83.2	85

Table 4.4: Evaluation for social anxiety with MIL algorithms

event-specific GMM training. Since our audio clips are weakly labeled, in the positive class audio clips a major proportion of the data does not indicate positive class. Hence, we cannot generate accurate class or event-specific GMM models using weakly labeled data. As shown in table 4.3, the BLSTM-MIL classifier achieves an F-1 score of 74.7% and 79% accuracy using I-vector features.

Our BLSTM-MIL implementation achieves F-1 scores of 56.82% and 55.55% using raw audio features (272 total) and audio words, respectively. We represent the generated audio words by one hot encoded vector of the size of the codebook. Hence, the feature dimension from each window for our BLSTM classifier is the size of the codebook. BLSTM-MIL performance is low with these high-dimensional discrete feature representations, which do not convey the audio-state-to-class (syntactic) relationship.

Emo2vec is a feature modeling technique that uses audio words (section 4.1.2) to generate vectors with the characteristics that, if two windows appear in a similar context (i.e., similar surrounding windows) for a specific vocal event (class), then the vectors will be similar. Since our audio clips are weakly labeled, the majority of the co-occurred windows are common for both positive and negative classes. Hence, generated Emo2vec vectors cannot convey the audio-state-to-class relationship. Emo2vec feature modeling reduces the feature space significantly. Hence, Emo2vec performs better than raw audio features and audio words, achieving an F-1 score of 72.3%.

Our NN2Vec approach generates low-dimensional continuous feature representation. NN2Vec vectors generated from the windows represent the state of audio from the respective windows and convey the audio-state-to-class (syntactic) relationship in its representation. This representation makes the classification task from weakly labeled audio clips easier (section 4.1.3). According to table 4.3, NN2Vec achieves an F-1 score 17% higher and 12% higher accuracy than those of the I-vector, the best baseline.

Comparison With MIL Baselines

This section discusses our evaluation of three MIL approaches using NN2Vec feature representation. Table 4.4 shows the results.

BLSTM-MIL implementation is similar to the approach described in section 4.4.1. The DNN-MIL classifier has three layers with [200, 200, 100] nodes and ReLU activation function, a 30% dropout rate, and one fully connected output dense layer [1] with sigmoid activation function. In this evaluation the mi-SVM implementation of MISVM toolkit [43] is used as the mi-SVM classifier.

Algorithm	F-1 Score	Accuracy
BLSTM-MIL	90.1	90
BLSTM	86.66	88.1
CNN-BLSTM	83.5	85.1
CNN	83.5	85.1
DNN	68.3	73

Table 4.5: Evaluation for social anxiety with supervised learning algorithms

Previous studies [198] have shown that DNN-based MIL approaches perform better than SVM-based implementations. In our evaluation the DNN-MIL approach achieves an F-1 score of 85%, which is only 2% higher than that of the mi-SVM approach. Our BLSTM-MIL approach achieves an F-1 score 5.6% higher than that of the DNN-MIL approach, the best MIL baseline.

Comparison With Supervised Learning Algorithms

This section compares BLSTM-MIL with supervised learning approaches using NN2Vec features. We consider as baselines the four most-evaluated supervised learning algorithms from the recent literature for human vocal event detection: BLSTM, CNN, CNN-BLSTM, and DNN. Table 4.5 shows the results. Given that input audio clips have variable lengths and the baselines require fixed-length input, input sequences were transformed to fixed length by zero padding. The following network parameter configurations were optimized by performing a grid search of the parameter values.

The CNN implementation has three convolution layers, each with 200 convolution kernels (temporal extension of each filter is 4), and ReLU activation function. The CNN uses a 20% dropout rate and max pooling windows of size 4 and down-scaling factor 2. Two fully connected dense layers [20,1] with sigmoid activation function are attached, which makes a binary classification decision. The network is trained with the mean squared error loss function and RMSprop optimization. The CNN implementation achieves an F-1 score of 83.5% and 85.1% accuracy.

The BLSTM classifier has three layers with [100, 100, 100] nodes, a 20% dropout rate, and two fully connected dense layers [20, 1] with sigmoid activation function. The network is trained with the mean squared error loss function and RMSprop optimization. Because BLSTM can learn from current audio state as well as use knowledge from relevant previous states, it performs better than other approaches for weakly labeled data. In our evaluation, BLSTM is the best baseline approach, achieving an F-1 score of 86.66% and 88.1% accuracy.

The CNN-BLSTM is a serial combination of CNN and BLSTM. Frequency variance in the input signal is reduced by passing the input through two convolution layers, each with 100 convolution kernels (temporal extension of each filter is 4), a 20% dropout rate, and ReLU activation function. The network uses max pooling windows of size 4 and down-scaling factor 2. After frequency modeling is performed, the CNN output (higher-order representation of input features) is passed to the BLSTM

layers. Two BLSTM layers [100, 100] and two fully connected layers [20, 1] are stacked at the end of the network architecture for the purpose of encoding long-range variability along the time axis and making the prediction.

The DNN implementation has three fully connected layers with [300, 300, 100] nodes and ReLU activation function, a 20% dropout rate, and one fully connected dense layer [1] with sigmoid activation function to make binary decisions. The DNN implementation achieves an F-1 score of 68.3%.

As shown in table 4.5, our BLSTM-MIL implementation (similar to section 4.4.1) achieves an F-1 score 3.9% higher than that of the best baseline (BLSTM) when both algorithms use NN2Vec vectors as features.

Our evaluations in section 4.4.1 and 4.4.1 show that the best baseline feature representation and supervised learning algorithm used in the literature are I-vector and BLSTM. Combining I-vector with BLSTM achieves an F-1 score of 71.4% and 76.2% accuracy. Hence, combining NN2Vec vector features with our BLSTM-MIL approach achieves an F-1 score 20.7% higher than that of the best baseline approach.

4.5 Evaluation: Depression

This section describes our evaluation of the NN2Vec and BLSTM-MIL approach for detecting depressed speakers on the DAIC-WOZ dataset (section 4.3.2). We performed all evaluations using leave-one-speaker-out cross-validation. To avoid overfitting, we randomly selected 30% of the audio clips for training the audio word codebook (section 4.1.2) and 30% of the audio clips to train the NN2Vec model (section 4.1.3) in each evaluation.

We performed a grid search on the model parameters window size, instance segment size, NN2Vec vector dimension, and audio codebook size from 500 ms to 10 seconds, 1 second to 60 seconds, 10 to 100, and 500 to 10000, respectively. The best parameter configuration was window size 2 seconds, instance segment size 25 seconds, a 20-dimensional NN2Vec vector, and audio codebook size 5000. The BLSTM-MIL classifier has two layers with [100, 100] nodes, a 20% dropout rate, and two fully connected dense layers [30, 1] with sigmoid activation function.

Feature	F-1 Score	Accuracy
NN2Vec	85.44	96.7
I-vector	70.1	86.54
Emo2vec	74.3	88.1
Audio word	51.2	79.1
Raw features	52.76	79.66

Table 4.6: Evaluation for depression with various feature representations

We evaluated the performance of our BLSTM-MIL implementation (section 4.2.1) of NN2Vec

against that of four baseline feature representations. As shown in table 4.6, our BLSTM-MIL approach using NN2Vec features achieves an F-1 score of 85.44% and 96.2% accuracy, whereas I-vector and Emo2vec achieve F-1 scores of 70.1% and 74.3%, respectively. Hence, NN2Vec achieves an F-1 score about 13% higher and 8% higher accuracy than those of Emo2vec, the best baseline feature representation.

Algorithm	F-1 Score	Accuracy
BLSTM-MIL	85.44	96.7
DNN-MIL	76.4	90.64
mi-SVM	66	84.1
BLSTM	77.1	91.4
CNN-BLSTM	71	87.6
CNN	68.8	85.1
DNN	56	80.86

Table 4.7: Evaluation for depression with baseline algorithms

Table 4.7 shows the results of our evaluation with MIL baselines and supervised learning baselines using NN2Vec features. Given that our supervised learning baselines require fixed-length input, audio input sequences were transformed to fixed-length by zero padding. The following network parameter configurations were optimized by performing a grid search of the parameter values. Our BLSTM-MIL achieves an F-1 score 10.5% higher than that of DNN-MIL, the best MIL baseline, and an F-1 score 9.7% higher than that of BLSTM, the best supervised learning baseline. The DNN-MIL classifier has three layers with [300, 300, 200] nodes and ReLU activation function, a 20% dropout rate, and one fully connected output dense layer [1] with sigmoid activation function. The BLSTM classifier has two layers with [200, 200] nodes, a 20% dropout rate, and two fully connected dense layers [50, 1] with sigmoid activation function.

We considered two of the most recent depression detection approaches [122, 100] evaluated on the DAIC-WOZ dataset as baselines. First, using I-vector features and Gaussian Probabilistic Linear Discriminant Analysis (G-PLDA) as the classifier [122] achieved an F-1 score of 57%. Second, DepAudioNet [100] encodes the temporal clues in the vocal modality using convolutional layers and predicts the presence of depression using LSTM layers. This serial combination of the CNN and the LSTM achieved an F-1 score of 52%. Hence, our BLSTM-MIL classifier using NN2Vec features achieves an F-1 score 33% higher than that of these other approaches.

4.6 Discussion

Identifying individuals high in social anxiety and depression symptoms using our NN2Vec features achieves F-1 scores 17% and 13% higher, respectively, than those of the best baselines (I-vector, section 4.4.1; Emo2vec, table 4.6). Moreover, combining NN2Vec features with our BLSTM-MIL

classifier achieves F-1 scores 20% and 33% higher, respectively, than those of the baselines (section 4.4.1 & 4.5).

In supervised learning, audio recordings are segmented into small fixed-length windows to train the CNN or CNN-BLSTM model. The labels of these windows are taken to be the same as the long audio clip-level labels. Hence, it is assumed that all the small windows in a positive long audio clip indicate high mental disorder symptoms. This, however, is not an efficient approach as it can result in a significant amount of label noise. Mental disorder symptoms in a long audio clip (segmented into a long sequence of windows) may be indicated by only a few seconds (a small subsequence of windows) of the clip, a fact ignored in assuming the label is strong. Due to the high label noise and limited training samples, convolution layers fail to generate effective higher-order representation of input features. To further support our statement, the DepAudioNet [100] approach applying the CNN-LSTM network using LLDs on the DAIC-WOZ dataset [193] achieved an F-1 score 8% lower than that obtained from using I-vector features and G-PLDA as the classifier [122] on the same dataset (section 4.5). By contrast, NN2Vec vectors map the audio from segmented windows into low-dimensional continuous feature space encoding the syntactic relationship between audio states (of windows), which facilitates a sequential classifier like BLSTM to effectively model the temporal properties in the speech signal (section 4.1.3). Hence, BLSTM and BLSTM-MIL classifiers perform better.

BLSTM networks are capable of learning and remembering over long sequences of inputs. This means that if a region (a small subsequence of windows) indicative of the positive class occurs in a long audio clip (long sequence of windows), BLSTM can pass that knowledge through hidden states. Studies [77] have shown that, as the sequence of windows becomes much longer for a limited training set, classifier performance starts to decline. Moreover, in a long positive weakly labeled audio clip, the portion of noise may significantly increase, making network optimization difficult. Hence, an MIL adaptation of BLSTM (BLSTM-MIL) performs better. In detecting speakers high in social anxiety symptoms (section 4.4.1), the window size was 1 second with 500 ms overlapping. Hence, input sequence size for the baseline BLSTM classifier (section 4.4.1) was 359 for 3-minute audio clips. The BLSTM-MIL classifier with input sequence size 25 (segment instance size 13) achieved an F-1 score about 4% higher than that of the best baseline BLSTM.

By contrast, detecting a depressed speaker (section 4.5) used a 2-second window size with 1 second overlapping. Hence, the input sequence size for BLSTM was 720 for 12-minute audio clips and achieved an F-1 score of 77.1%. In this evaluation, the BLSTM-MIL classifier with input sequence size 24 (segment instance size 25) achieved a 10% higher F-1 score. Hence, these evaluations (section 4.4.1 and table 4.7) show that as the input sequence size (audio clip length) increases, BLSTM-MIL performs increasingly better than BLSTM.

The ability to identify symptomatic individuals from their audio data represents an objective

indicator of symptom severity that can complement health-care providers' other assessment modalities and inform treatment. Moreover, because vocal analysis does not require extensive equipment and is readily accessible (speech is ubiquitous in natural settings), nonintrusive (it does not require a special wearable monitor), and not burdensome (it does not require additional assessment time or client responses), it is scalable, which is important given the vast number of people with social anxiety and depression who receive no help [37]. Additionally, the unique source of the data (animated virtual clinical interview) for the depression detection study further supports the possibility of using vocal analysis as a remote assessment tool and one that may eventually be possible to administer via artificial intelligence. This is especially exciting in light of the difficulty in identifying and disseminating care to people facing considerable barriers to seeking treatment for social anxiety [127] and depression [45].

Further, the ability to detect individuals high and low in social anxiety symptoms opens new possibilities for assessment, treatment, and prevention. Implementing vocal analysis with mobile technologies (e.g., smartphones) would give health-care providers an objective marker of clients' anxiety as it unfolds outside of the treatment setting, and combining this with other data (e.g., location, actigraphy) could help clarify the antecedents and consequences of clients' anxious states. Providers could even collect such idiographic time series data from clients before treatment to understand each client's dynamic processes and personalize treatment from the start [50]. Moreover, pairing passive outcome monitoring with mobile interventions (e.g., skills training apps) would enable the timely delivery of just-in-time interventions that may offer relief and efficiently promote skills acquisition and generalization. Along these lines, detecting disorder symptoms from audio data may one day be used to identify changes in speech that suggest a person may be transitioning to a higher-risk state and could benefit from preventive services to avoid the worsening of symptoms.

The present study has several limitations related to sampling. First, we used an analog sample of people high versus low in social anxiety symptoms for whom no formal diagnoses of social anxiety disorder had been established. Second, we analyzed speech audio data from only one situation (a speech stressor task), so future work would benefit from sampling speech from a wider range of both social and nonsocial situations to determine the boundaries of the models' predictive validity.

Finally, we wish to emphasize that implementation of our approach, even if designed to support health-care providers, must include the informed consent of clients, who should be allowed to discontinue the monitoring at any time, and robust privacy protections. It is important to note that our approach does not use the semantics (transcribed text) of the client's speech and that the proposed feature extraction is irreversible (section 4.1.2), thereby ensuring clients' privacy. Any feedback provided to the client about increases in symptoms would ultimately be paired with treatment resources or other services (e.g., interventions) that the client can use to seek relief. Further, future research is needed to evaluate the feasibility, acceptability, and safety of our approach before providers implement the approach on a large scale in the community.

4.7 Summary

Mental health problems, such as depression and social anxiety disorder, are often under-diagnosed and under-treated, in part due to difficulties identifying and accessing individuals in need of services. Current assessments for these disorders are typically based on client self-report and clinical judgment and therefore are subject to subjective biases, burdensome to administer, and inaccessible to clients who face barriers to seeking treatment. Objective indicators of depression and social anxiety would help advance approaches to identification, assessment, prevention, and treatment. This chapter presents a weakly supervised learning framework for detecting symptomatic individuals from long speech audio data. Specifically, we present a novel feature modeling technique named NN2Vec that identifies and exploits the inherent relationship between vocal states and symptoms/affective states. In addition, we present a new MIL adaptation of the BLSTM classifier, named BLSTM-MIL, to comprehend the temporal dynamics of vocal states in weakly labeled data. We evaluated our framework on 101 participants' spontaneous audio speech data weakly labeled with speakers high in social anxiety. Our NN2Vec and BLSTM-MIL approach achieved an F-1 score of 90.1% and 90% accuracy in detecting speakers high versus low in social anxiety symptoms. This F-1 score is 20.7% higher, than those of the best baselines. To our knowledge, this study is the first to attempt such detection using weakly labeled audio data [159]. Using audio clips from virtual clinical interviews, our approach also achieved an F-1 score of 85.44% and 96.7% accuracy in detecting speakers high versus low in depressive symptoms. This F-1 score is 33% higher than those of the two most recent approaches from the literature.

Chapter 5

Behavioral Vocal Events

In detection of verbal behavioral events through speech, two factors are important: the lexical content (i.e., spoken words) and acoustic variation. When a speaker expresses a verbal event while adhering to an inconspicuous intonation pattern, listeners can nevertheless perceive the information through the lexical content (i.e. words). On the other hand, some verbal event conveying sentence structures share the same lexical representation with other general statements. Hence, verbal behavioral events are not detectable only from contextual data. Detection of these verbal events also depends on speaker's tone or acoustic features. This chapter presents DAVE, a comprehensive set of verbal behavioral event detection techniques that extracts textual features from transcribed speech as well as extracts acoustic signal features from respective speech portion. Both of the textual and acoustic signal features are used to discriminate the verbal behavioral events from others.

The medical community has defined the Cohen-Mansfield Agitation Inventory [35] which specifies approximately 28 agitated behaviors for identifying whether a person is suffering from agitation. DAVE is a set of approaches that addresses the challenges of real time monitoring and recording the 5 most important of the vocal agitation metrics of the Cohen-Mansfield Inventory. This includes cursing, constant unwarranted request for help, making verbal sexual advances, asking constant questions and talking with repetitive sentences.

Another reason for choosing these 5 vocal events is that, the scope of applications that can use DAVE lies well beyond monitoring agitated elderly suffering from dementia. Detection of atypical vocal events are useful for online video sharing sites such as *Youtube* and movies, where providers and users are able to detect objectionable content such as cursing, sexual advances, etc to impose restrictions (e.g., for children). Detection of asking for help and questions can improve several human computer interaction (HCI) systems such as: automated customer service interaction systems, smart classrooms, etc. Also, some of the vocal events such as: asking for help, verbal sexual advances, and cursing are important for home safety and all of the 5 events are important for home health care. Hence, we have developed novel solutions based on various combinations of features.

The main contributions of this study are:

- An automatic and comprehensive set of techniques developed for detecting 5 verbal agitations based on both extending various algorithms and combining acoustic signal processing with three different text mining paradigms.
- None of the previous state of the art solutions has addressed: asking for help and verbal sexual advances. In this study we are the first to show that detection of these two vocal events depends both on the acoustic signal processing and the semantics of the speech. To understand the semantics of speech we employ statistical text data mining techniques. Using such a combined feature set we achieve a detection accuracy of 93.45% for asking for help and a detection accuracy of 91.69% for verbal sexual advances.
- Cursing is difficult to detect because many such words have multiple meanings. We have used a modified version of the adapted Lesk algorithm [17] which considers a word's sense, to detect curse words with multiple ambiguous meanings. Using this approach we have detected cursing with 95.6% accuracy.
- We are the first to evaluate a large combination of acoustic, tf-idf and language model features to detect questions from English speech data, and achieved 89.68% accuracy.
- Repetitive sentences from an agitated patient are not precisely repetitive. We have addressed the issue of skipping or adding multiple words in sentences by using a modified version of the prefixSpan algorithm [134] and achieved 100% accuracy.
- We have evaluated DAVE on 34 real agitated elderly (age varies from 63 to 98 years) dementia patients across 16 different nursing homes and achieved 90%, 88.1%, 94% and 100% precision for verbal events: asking for help, questions, cursing and asking repetitive sentences, respectively. Here we solve the challenge that dementia patients mumble, speak in low volume and don't articulate words well. (Section 5.3).
- To show it's generalizability to different domains and for the healthy population, we have evaluated DAVE on movies, *Youtube* clips, the Tatoeba website speech clips [2] with acted and real vocal events, using audio clips from controlled experiments and from real homes. We show accuarcy in the 90-100% range. (Section 5.2).

5.1 Design of DAVE

DAVE consists of two categories of solutions: detection of *(asking for help, verbal sexual advances and questions)* uses a combination of acoustic and bag-of-word textual features and detection of *(cursing and using repetitive sentences)* uses textual features only, shown in Figure 5.1.

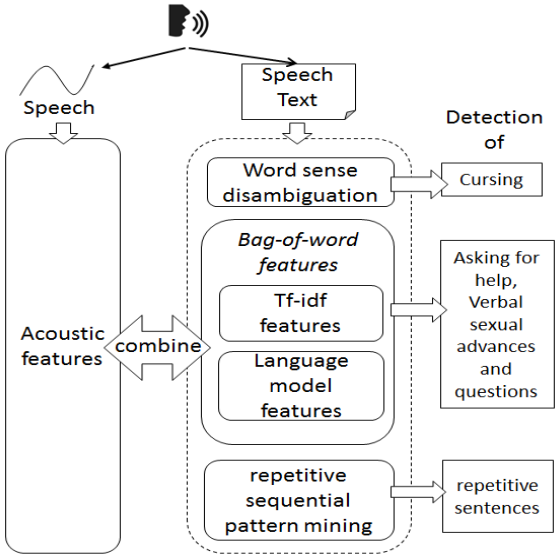


Figure 5.1: Block Diagram of DAVE

5.1.1 Textual and Acoustic Features

Acoustic analysis is significant to detect *asking for help, verbal sexual advances* and *questions*, since studies [213] have shown that, human behaviors are consistent with specific conscious and unconscious emotion concepts. But, relying only on acoustic signal processing might result in inaccuracy since asking for help, verbal sexual advances and questions rely heavily on semantics of speech data. Hence, our solution combines acoustic signal processing with textual inference to detect these three verbal events. In the following subsections we discuss the textual and acoustic features evaluated to detect *asking for help, verbal sexual advances* and *questions*.

Text Features

The Bag-of-word representation is widely used in text data analysis. Two of the most widely used bag-of-word representation models are the tf-idf vector model and language models where terms are assumed to be unrelated, in the sense that each term is considered to be an atomic unit of information. DAVE considers converted speech text as a document and extracts bag-of-word features from that document which represent the textual concept of speech. In our solution, a combination of unigram and bigram words are used as terms.

Tf-idf Features tf-idf stands for 'Term Frequency, Inverse Document Frequency' which is a way to score the importance of terms in a text document based on how frequently they appear in that text document and across multiple text documents, where each text document is represented as a

text vector and each dimension corresponds to an individual term. The value of a term in this vector shows how important that term is to represent that text document [164].

We represent text portions of our converted speech text using a text vector that captures the relative importance of the terms in the text. The value of a term in our text vector representation is calculated using *tf-idf weighting*. Intuitively, if a term appears frequently in a text document, it is important. Since that relation is not linear, as shown by equation 5.1 we have used *sublinear tf scaling* to calculate tf where $C(t, d)$ is the frequency of occurrences of term t in text document d. And idf measures how important a term is in an overall sense. While computing tf, all terms are considered equally important. However certain terms, such as "is", "of", and "that", may appear many times, but have little importance. Thus idf is used to lower the emphasis of frequent terms while scaling up the rare ones, which is computed using equation 5.2 where N is the total number of documents (sentences) in the training corpus and $DF(t)$ is the total number of documents containing term t. Hence, $tf - idf = tf \times idf$. We represent the text portion from which we want to detect these three verbal events using this text vector representation, and use the term weights from the text vector representation as textual features.

$$tf_t = \begin{cases} 1 + \log C(t, d) & \text{if } C(t, d) > 0 \\ 0 & \text{else} \end{cases} \tag{5.1}$$

$$idf_t = 1 + \log(\frac{N}{DF(t)}) \tag{5.2}$$

Language Model Features In utilizing language models, our solution uses the ratio between the log-likelihood of the sentence with respect to the 'verbal event (asking for help, verbal sexual advances or questions) language model' and the log-likelihood of the sentence with respect to the 'non-verbal event language model' as language model features. This log-likelihood ratio (LLR) is computed as:

$$LLR(S) = \log(\frac{P(S|verbaleventLanguageModel)}{P(S|nonverbaleventLanguageModel)}) \tag{5.3}$$

Here, $P(S|C)$ is the conditional probability of sentence S given class C, where $C \in$ ('verbal event Language Model' or 'non-verbal event language model').

We have explored both the unigram and bigram language models. The language models are computed with maximum likelihood estimation. Equation 5.4 and 5.5 show the calculation of unigram model and bigram language model probability where $N(T)$ is the frequency of the term $T \in$ (unigram or bigram) in the training corpus.

$$P^{Uni}(w_i) = \frac{N(w_i)}{\Sigma_{j \in allwords} N(w_j)} \tag{5.4}$$

$$P^{Bi}(w_i|w_{i-1}) = \frac{N(w_{i-1}, w_i)}{N(w_{i-1})} \tag{5.5}$$

In the case of the unigram language model, $P(S|C)$ is calculated by equation 5.6 and the bigram language model is calculated by equation 5.7, where $S = w_1, w_2...w_L$.

$$P^{Uni}(S|C) = \prod_{i=1....L} P(w_i|C) \tag{5.6}$$

$$P^{Bi}(S|C) = P(w_1|C) \prod_{i=2....L} P(w_i|w_{i-1}C) \tag{5.7}$$

It is important for language models to attribute a non-zero probability to the words or n-grams that are not seen in a set of training documents (training corpus). To avoid zero probability in calculating the probabilities we used following smoothing methods. These smoothing methods subtract a very small constant from the probability of seen events and distribute it over all seen and unseen events.

Additive Smoothing: Equation 5.8 shows the computation of additive smoothing for the unigram language model of class C where δ is the smoothing parameter. $N(w_i|C)$ represents the frequency of word w_i in the training corpus. $|C|$ is total word count and $|V|$ is the vocabulary size of the training corpus.

$$P^{AS}(w_i|C) = \frac{N(w_i|C) + \delta}{|C| + \delta|V|} \tag{5.8}$$

Linear Interpolation Smoothing: This smoothing method use (N-1)gram probabilities to smooth N-gram probabilities. Equation 5.9 shows the computation of linear interpolation smoothing for the bigram language model of class C where λ is the smoothing parameter to be determined and $P^{AS}(w_i|C)$, $P^{Bi}(w_i|w_{i-1}C)$ are computed using equation 5.8,5.5 respectively.

$$\begin{aligned} &P^{LIS}(w_i|w_{i-1}C) = \lambda P^{Bi}(w_i|w_{i-1}C) \\ &+(1-\lambda)P^{AS}(w_i|C) \end{aligned} \tag{5.9}$$

Absolute Discounting Smoothing: Equation 5.10 shows the computation of the absolute discounting smoothing for the bigram language model of class C where δ is the smoothing parameter, $N(w_i)$ is the frequency of word w_i, S is the number of seen word types occur after w_{i-1} in the training corpus and $P^{AS}(w_i|C)$ that is computed with equation 5.8.

$$\begin{aligned} &P^{ADS}(w_i|w_{i-1}C) = \tfrac{max(N(w_{i-1},w_i)-\delta,0)}{N(w_{i-1})} \\ &+\tfrac{\delta S}{N(w_{i-1})} P^{AS}(w_i|C) \end{aligned} \tag{5.10}$$

Hence, we have three language model features (log-likelihood ratios) for each verbal event; One each from the unigram language model with additive smoothing, the bigram language model with linear interpolation smoothing and the bigram language model with absolute discounting smoothing.

Acoustic Signal Features

Since, human behaviors remain consistent with specific emotion concepts [213], our goal is to extract acoustic features to represent those emotional concepts that are depicted through their tone of speech. The arousal state of the speaker affects the overall energy, energy distribution across the frequency spectrum, and the frequency and duration of pauses in a speech signal. Hence, the primary continuous acoustic features: *energy and pitch* are used as features in our analysis.

Another important continuous feature is the fundamental frequency *(F0)*, that is produced by the pitch signal, also known as the glottal waveform, which carries speaker tone information because of its dependency on the tension of the vocal folds and the subglottal air pressure.

The harmonics to noise ratio (HNR) in speech provides an indication of the overall aperiodicity of the speech signal. Breathing and roughness are used as parameters for speech analysis and they are estimated by HNR. There is significantly higher HNR in the sentences expressed with anger than the neutral expressions. *Zero crossing rate* is a measure of number of times in a given time interval/frame that the amplitude of the speech signals passes through a value of zero. There is a strong correlation between zero crossing rate and energy distribution with frequency and a reasonable generalization is that if the zero crossing rate is high, the speech signal is unvoiced. Also, *the voicing probability* computed from the *ACF* indicates an acoustic signal is from speech or non-speech. *MFCC* features are the means by which spectral information in the sound can be represented. Here the changes within each coefficient across the range of the sound are examined. These features take human perception sensitivity with respect to frequencies into consideration.

Hence, in our acoustic analysis the low-level descriptors extracted from small frames are: *zero-crossing-rate (ZCR) from the time signal, root mean square (RMS) frame energy, pitch frequency (normalised to 500 Hz), harmonics-to-noise ratio (HNR) by the autocorrelation function, the fundamental frequency computed from the Cepstrum and mel-frequency cepstral coefficients (MFCC) 1-12 in full accordance to HTK-based computation.* To each of these, the *delta coefficients* are additionally computed. Next the 12 functionals: mean, standard deviation, kurtosis, skewness, minimum and maximum value, relative position, and range as well as two linear regression coefficients with their mean square error (MSE) are applied on a chunk of small frames.

Combination of Features Used in Solution

Various combinations of features from sections 5.1.1 & 5.1.1 are evaluated for each of the verbal events: asking for help, verbal sexual advances and questions. Through our extensive evaluation we conclude to use a combination of acoustic and all 3 language model features as input to detect verbal events: verbal sexual advances and questions. Also, asking for help is detected using a combination of all the bag-of-word features with acoustic features. Our solution is shown in Figure 5.2.

Detection classifier

Features extracted from both acoustic signals and converted textual data are used as input for a detection classifier. We have used three separate binary classifiers to detect each of these 3 verbal events, where class labels are positives and negatives. We have explored the NaiveBayes, K-nearest neighbor and SVM classifiers as a detection classifier for each of these verbal events (see section 5.2 for the evaluation).

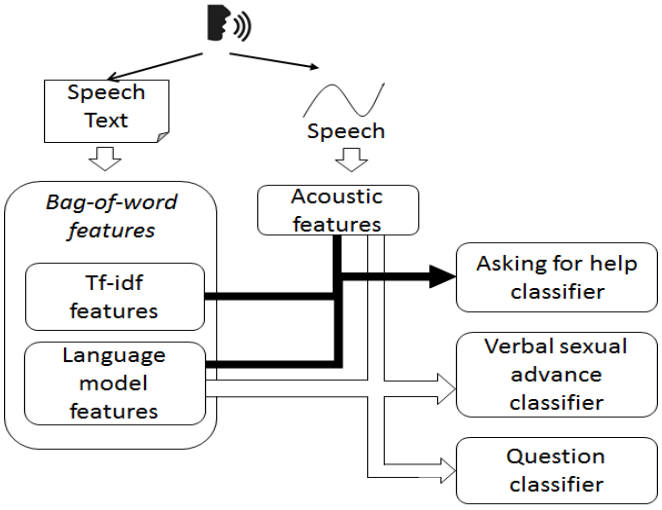


Figure 5.2: Feature combinations to detect verbal events

5.1.2 Using Text Only

Detecting Agitated Event: Cursing

To detect *cursing*, we have built a word dataset which contains 165 most used curse words. The words in the converted text are matched with the words of the curse word dataset. If a match is found, it is considered that the curse might have occurred, but we must check the sense in which the word was used. Within these curse words some of them have multiple meanings or word senses. Such as word: 'dog' can be used to describe a pet, also, it can be used as a curse word. Since, linguistic content of cursing does not contain compound concepts, to address this challenge we performed word sense identification analysis to detect the latent semantic meaning or word sense of these curse words (instead of text document representation features in section 5.1.1).

There are several word sense analysis approaches, such as knowledge based methods, supervised methods, and semi-supervised methods. Supervised and semi-supervised approaches need a large dataset and they identify semantic meanings of a word in a specific domain. Our goal is to identify if a word is used as a curse word or not in a generic context. Also, due to the lack of large curse

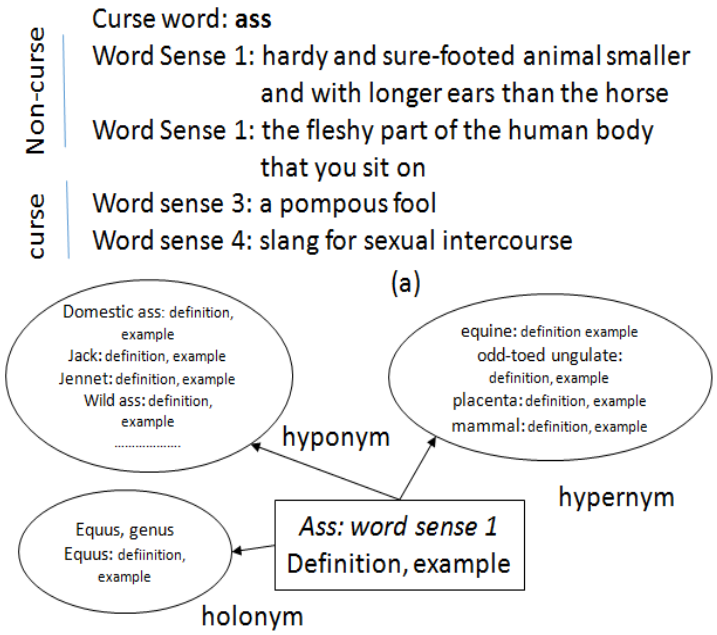


Figure 5.3: WordNet wordsenses for word:ass and relation of one wordsense of word:ass with synset words

word datasets we have developed a knowledge based approach for our cursing detection. We have used a modified version of the Adapted Lesk Algorithm [17], which uses Wordnet [114] to detect a word sense using the context of neighbor words. WordNet [114] is a lexical database for the English language that groups English words into sets of synonyms called synsets. It provides short definitions and usage examples, and stores relations such as *hypernyms, hyponyms, meronyms, troponyms* etc. among these synonym sets.

To detect the latent word sense of each of the curse words with multiple meanings from the converted text, we define K neighbor context words around the target curse word. For each word in the selected context, our algorithm looks up and lists all the possible senses of two parts of speech: noun and verb. For each word sense our algorithm takes into account its own gloss or definition and examples provided by WordNet [114], and the gloss and examples of the *synsets* that are connected to it through *hypernym, hyponym, meronym or troponym* relations to build an enlarged context for that word sense. All the enlarged contexts for each word sense of all these context words are compared with the enlarged context of each of the word senses of the targeted curse word. The enlarged word sense contexts that overlap most with the enlarge context of all the word senses of neighbor context words of the targeted curse word is the word sense of the targeted curse word.

As an example, suppose a curse word:‘ass’ occurs in a converted speech text. Figure 5.3 (a) shows the 4 word senses of the curse word:‘ass’ extracted from WordNet, where word sense 1 &

2 are categorized as 'non curse senses' and word sense 3 & 4 are categorized as 'curse senses'. DAVE builds a enlarged context set for all the word senses considering the *hypernyms, hyponyms, meronyms, troponyms* relationship in synset. For example, Figure 5.3 (b) shows the relationships of *'wordsense 1'* of the curse word 'ass' in synset provided by WordNet. All the words in the definition and examples of word sense 1 and of related words shown in figure 5.3 (b) are included in the enlarged context set for word sense 1. Hence, the enlarged context set for *'non curse category'* and *'non curse category'* is the union of the enlarged context sets of all the word senses included in that respective category.

To detect cursing we have used WordNet instead of a dictionary since, while traditional dictionaries are arranged alphabetically, WordNet is arranged semantically where each word is connected with words in its synset based on various semantic relations.

Detecting Repetitive Sentences

To detect repetitive sentences or questions from text we performed indexing and give unique IDs to each of the words in the text data. Then we convert the words to their corresponding IDs in the text. We modified prefixSpan [134] to find the repetitive subsequences which occurred a minimum of T times in the converted word ID sequence. Since, repetitive sentences from an agitated patient may not be exactly the same; we identify that the repetition of sequence of words has occurred if word sequences match with a maximum of s_n number of words skipped. That means, if s_n is 2, "I eat chocolate" matches with "I eat too many chocolate" but it is not matched with "I eat too too many chocolate". Our prefixSpan [134] modification limits the expansion of search space into further branches using the knowledge of minimum number of repetitions required and maximum number of allowed word skips. Suppose a converted sequential word ID representation of a text is $< W_1W_2W_3W_1W_2W_4W_3W_5 >$ where each W_i is the unique word ID of the ith unique word of the text. If we consider T as 2 and s_n as 1, the search space of our algorithm is shown in figure 5.4. Here, the search space is divided into 5 branches, one for each of the unique word IDs. This growth-based approach of finding sequential words grows larger by dividing the search space and focusing only on the subspace potentially supporting further growth. Unlike traditional apriori based approaches which perform candidate generation and test, this approach does not generate any useless candidate sequences. Also, two word sequences extracted from the left-most branch in figure 5.4 are combined since, sequence $< W_1W_2 >$ is a subset of sequence $< W_1W_2W_3 >$. Hence, our resultant repetitive word sequences are: $< W_1W_2W_3 >$, $< W_2W_3 >$. Studies [134] have shown that, in the average case the growth based approaches for sequential pattern mining perform up to 40% faster and uses about 0.1 times the memory for computation, compared to other apriori based approaches. For DAVE we use $s_n = 3$.

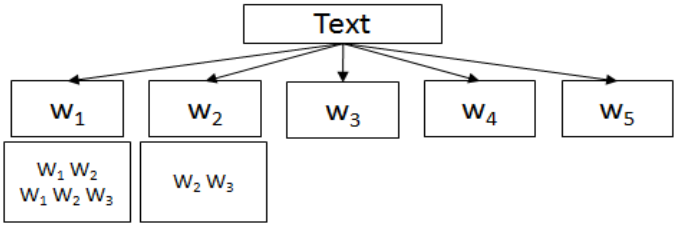


Figure 5.4: Search space of modified version of prefixSpan

5.2 Evaluation on Healthy People

Section 5.2.1 describes the experimental setup and datasets used from movies, Youtube, Taboeba website and our own data collection. Using this data, the next three sections show the evaluation for vocal events that require both bag-of-word textual features and acoustic information (section 5.2.2), that require word sense disambiguation from text information (section 5.2.3), and that require repetitive sequential pattern mining from text information (section 5.2.4), respectively. Through these evaluations we find which combinations of features and approaches provide higher detection accuracy for respective verbal events in generic domains.

5.2.1 Experimental Setup - Preliminaries

Acoustic Pre-processing

For completeness, this section describes the noise filtering and the conversion of audio to text.

Filtering and Removing Noise: The first step for pre-processing is to remove unvoiced audio segments using zero crossing rate (ZCR) [16]. To capture the pause in spoken sentences, the detected voiced segments are lengthened by 1 second on both sides. If another consecutive voiced segment starts within the lengthened 1 second segment portion, both the voice segments are merged into one.

Noises which are out of human voice frequency range were removed using a bandpass filter with a low frequency of 80Hz and a high frequency of 3000Hz. Hiss, hum or other steady noises were reduced using a spectral noise gating algorithm [7].

Converting Audio to Text: Since all of our solutions require text, we require audio to text conversion of the sound clips. We have used Dragon NaturallySpeaking [1] which is a speech recognition and transcription system.

Textual Data for Training Lexical Models

We used separate training corpuses to compute 'Document Frequency' of unigram and bigram words and language models for each of the verbal events: asking for help, verbal sexual advances

and questions. We learned two language models corresponding to each of the verbal events (asking for help, verbal sexual advances and questions) while one detects the presence of that verbal event, other detects the absence. These language models have the purpose of representing the main word sequences that occur in an utterance from respective verbal events rather than other events.

The wiki talk pages [4] consist of threaded posts by different authors about a particular wikipedia entry. While the sentences from these posts lack certain properties of spontaneous speech, they are more conversational than articles. Tatoeba website [2] contains a large collection of human spoken sentences in text and audio. Also, the urbandictionary website [3] has a large collection of human spoken sentences performing verbal events: verbal sexual advances and cursing. We labeled sentences from wiki talk posts, tatoeba and urbandictionary websites to include them in our training corpuses.

To achieve lexical characteristics of spontaneous verbal event utterances we conducted a survey of 21 volunteers where participates were asked what they will say to perform our targeted verbal events in different random scenarios and included the responses in respective training corpuses. Table 5.1 shows the number of sentences in training corpuses for each of the verbal events: asking for help, verbal sexual advances and questions.

Training corpus	Verbal events		non-verbal events
	From survey	From other sources	
Asking for help	144	856	3000
Verbal sexual advances	98	601	3000
Questions	335	1165	5000

Table 5.1: Number of sentences in training corpuses

Data for Verbal Event Detection Evaluation

Since there is no existing available dataset for asking for help, verbal sexual advances, questions, cursing and talking with repetitive sentences we had to create our own dataset for training and evaluation. We have collected verbal speech data from 6 individuals, whose ages varies from 21 to 30. There were 4 females and 2 males. We also, collected human spoken sentences audio clips from Tatoeba website [2]. To enrich our dataset we have included audio clips from movies and real *Youtube* videos where people are performing our targeted verbal events. Table 5.2 shows the number of audio clips for evaluation for each of the 5 verbal events. These clips have lengths varying from 2 to 20 seconds, containing 1 to 23 words. To evaluate curse detection from audio we have collected 260 clips from movies, among 137 of them people used 'cursing' in their conversation. Among these 137 'cursing' events, 91 of them have curse words which have a single meaning, and 46 of them have multiple meanings. Also, 50 audio clips have multiple meaning 'curse' words with non-curse meanings.

Verbal event	Number of clips
Asking for help	260
Verbal sexual advances	165
Questions	400
Cursing	260
Repetitive sentences	80
Others	500

Table 5.2: Number of clips for evaluation of verbal event detection

Pre-processing of Textual Data

Stop words are usually the most frequent words including articles, auxiliary verbs, prepositions, conjunctions and they do not provide additional improvement for textual similarity analysis. We have created a customized stop-word list for verbal events: asking for help, verbal sexual advances and questions, and created our vocabulary set with corresponding *idf* values from the training converted text set. We used Porter steaming to reduce inflected words to their base form and normalization to remove punctuation marks, and converted words to lower case in our process of vocabulary building. After this pre-processing the vocabulary size for asking for help, verbal sexual advances, and questions are 214, 178, 658, respectively.

5.2.2 Verbal Events: Combination of Acoustic and Text Data

For each of the verbal events: asking for help, verbal sexual advances and questions we have trained a binary class classifier. Since, binary classifiers do not work well when trained with imbalanced data sets: new instances are likely to be classified as the class that has more training samples. In order to avoid this over-fitting problem, we chose to resample the dataset by keeping all clips for respective verbal events and randomly extracting subsets of clips of the same size (sampled from other 5 categories of table 5.2). In the following section we evaluate how a verbal event detection classifier performs using only acoustic features (section 5.2.2), then we evaluate how the detection accuracy changes by adding textual features into the classifier (section 5.2.2). All the evaluations were done using 10-fold cross validation with 33.33% of the data as test data. We used accuracy, precision, and recall as our detection performance evaluation metrics.

Acoustic Features

We tested the NaiveBayes, K-nearest neighbor and SVM classifiers using the acoustic features discussed in section 5.1.1. Table 5.3 shows the evaluation of these three verbal events using only acoustic features. As we can see, the accuracy is ranging between 66% to 82.5%.

Event	Classifier	Accuracy	Precision	Recall
Asking for help	NaiveBayes	71.72	0.707	0.718
	K-nearest neighbor	80.314	0.803	0.805
	SVM	79.031	0.811	0.79
Verbal sexual advances	NaiveBayes	73.35	0.78	0.734
	K-nearest neighbor	81.43	0.813	0.814
	SVM	82.54	0.824	0.826
Questions	NaiveBayes	66.093	0.719	0.661
	K-nearest neighbor	81.97	0.821	0.82
	SVM	82.22	0.829	0.822

Table 5.3: Evaluation with acoustic features

Combination of Acoustic-Textual Features

According to Table 5.3 the best detection accuracy using acoustic features alone are 82.54%, 80.31% and 82.22% for verbal events: verbal sexual advances, asking for help and questions. To achieve higher accuracy, we introduce textual features which represent the semantic understanding of speech while expressing these verbal events.

The studies related to automatic speech recognition systems have to additionally take into account the speech recognition errors which get more frequent for poor sound qualities and on spontaneous speech, and can highly decrease the classification performance. Hence, the classifier evaluations are carried out using features stemming from:

- manual transcriptions - to study the classifier's maximum performance, obtainable only in ideal conditions (i.e. with perfect transcripts)
- automatic transcriptions (obtained with Dragon speech recognizer) - to study the performance under real conditions

Sections 5.2.2, 5.2.2,and 5.2.2 are devoted to the evaluation of verbal event detection using a combination of acoustic and tf-idf features, using a combination of acoustic and language model features and a combination of acoustic and all bag-of-word features.

Combination: Acoustic and Tf-idf Features In this work two types of tf-idf features: unigram and bigram were extracted from transcribed speech text. For all the evaluations shown in this section, the SVM classifier gave higher accuracy, hence only the results obtained with the SVM classifier are presented here. Figure 5.5 and 5.6 shows the evaluation combining acoustic with unigram and all (unigram and bigram) tf-idf features, respectively with both manual and automatic transcription. According to these evaluations detection accuracy increases up to 91.88%, 89.44% and 88.92% for verbal events: asking for help, verbal sexual advances and questions with both unigram and bigram textual features extracted from manual transcription. Accuracy decreases by 0.28%, 1.56% and 1.1%

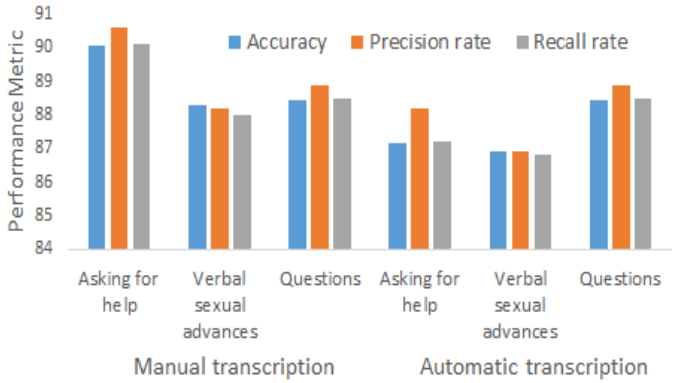


Figure 5.5: Evaluation with acoustic and unigram tf-idf features

when tf-idf features are extracted from automatic speech transcription. According to our evaluation most important tf-idf terms (higher tf-idf values) for questions and asking for help event detection are *wh-words* (why, who, which, what, where, when, and how) and 'help', 'please', 'need', 'can you', 'will you' , etc. respectively. Transcription error rate for these simple words are low for Dragon NaturallySpeaking software, hence decrease of detection accuracy for: asking for help and questions detection is lower compared to verbal sexual advances detection.

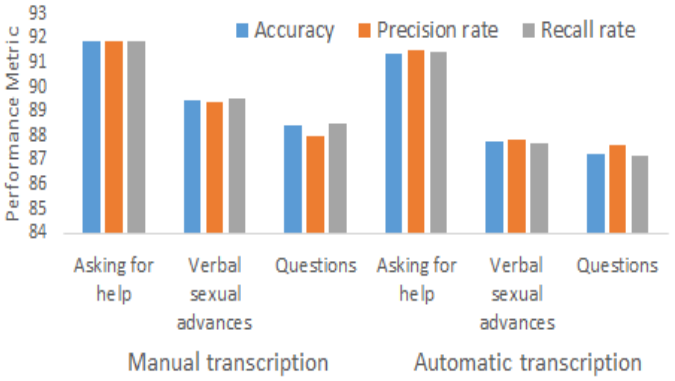


Figure 5.6: Evaluation with acoustic, unigram and bigram tf-idf features

As shown in figure 5.5, with manual transcription questions detection accuracy (88.45%) is higher using combination of acoustic and unigram tf-idf features. Hence, we conclude that questions detection perform better using a combination of acoustic and unigram tf-idf features, where other two event detections perform better using acoustic features in addition to both unigram and bigram tf-idf features.

Combination: Acoustic and Language Model Features In our evaluation three language model features: 'unigram log-likelihood ratio', 'log-likelihood ratio of bigram language models with linear interpolation smoothing' and 'log-likelihood ratio of bigram language models with absolute discount smoothing' are extracted from speech text. For all the evaluations shown in this section, the SVM classifier gave higher accuracy, hence only the results obtained with the SVM classifier are presented here.

Table 5.4 shows the evaluation using various combinations of language model features in addition to acoustic features to detect verbal events: asking for help, verbal sexual advances and questions using manual transcription. According to this evaluation, all 3 language model features in addition to acoustic features used as input provide higher accuracy for all 3 verbal events. Hence, in the later sections of this chapter the term 'using language model features' will be referred to as using all 3 of the language model features.

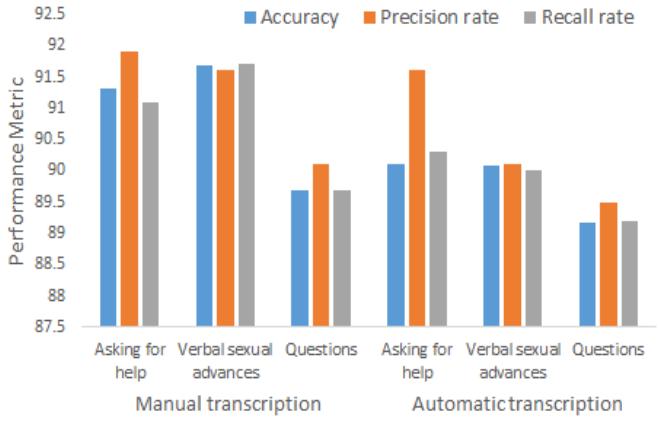


Figure 5.7: Evaluation with acoustic and language model features

Figure 5.7 shows the evaluation metrics using acoustic and language model features extracted from manual and automatic transcription. Using manual transcription, the highest accuracy for asking for help is 91.36%, which is lower compare to detection using acoustic and tf-idf features as shown in Figure 5.6. On the contrary, accuracy for verbal sexual advances and questions detection increases to 91.69% and 89.68%, respectively using a combination of acoustic and language model features as input for the classifier. Detection accuracy decreases by 0.8%, 1.74% and 0.05% for verbal events: asking for help, verbal sexual advances and questions, respectively when textual features are extracted from automatic speech transcription, which complies with our evaluation in section 5.2.2.

Combination: Acoustic and Textual Features Tables 5.5 and 5.6 show the evaluation metrics of verbal events: asking for help, verbal sexual advances and questions detection using acoustic

Event	Language model features	Acoustic features	Accuracy	Precision	Recall
Asking for help	Additive smoothing	All	87.95	0.894	0.88
	Linear interpolation smoothing	All	87.958	0.895	0.885
	Absolute discounting smoothing	All	87.96	0.9	0.88
	All 3 features	All	91.1	0.919	0.911
Verbal sexual advances	Additive smoothing	All	90.943	0.909	0.91
	Linear interpolation smoothing	All	90.944	0.91	0.91
	Absolute discounting smoothing	All	90.944	0.91	0.91
	All 3 features	All	91.6981	0.916	0.917
Questions	Additive smoothing	All	88.206	0.885	0.882
	Linear interpolation smoothing	All	88.24	0.886	0.884
	Absolute discounting smoothing	All	87.96	0.882	0.88
	All 3 features	All	89.68	0.901	0.897

Table 5.4: Evaluation of various combinations of language model features in addition to acoustic features with manual transcription

features in addition to both bag-of-word (tf-idf and language model) textual features extracted from manual and automatic transcriptions, respectively. As shown in Tables 5.5 and 5.6 combining all textual and acoustic features we achieve up to 93.45% and 91.36% accuracy for asking for help detection using manual and automatic transcription. According to Tables 5.5 & 5.6, and Figure 5.7 the highest achieved accuracy of verbal sexual advances and questions detection using a combination of all textual and acoustic features are similar to detection using acoustic and language model features only, for both manual and automatic transcription.

Hence we conclude that, among all the combinations of features we have evaluated, a combination of acoustic and language model features are sufficient to detect verbal events: verbal sexual advances and questions, where to achieve higher accuracy for asking for help we need a combination of all the

Event	Classifier	Accuracy	Precision	Recall
Asking for help	NaiveBayes	80.36	0.801	0.804
	KNN	91.36	0.915	0.914
	SVM	93.45	0.934	0.935
Verbal Sexual Advances	NaiveBayes	81.13	0.846	0.811
	KNN	87.15	0.874	0.872
	SVM	91.69	0.916	0.917
Questions	NaiveBayes	71.74	0.755	0.717
	KNN	87.96	0.88	0.89
	SVM	89.697	0.9	0.897

Table 5.5: Evaluation with combined features with manual transcription

Event	Classifier	Accuracy	Precision	Recall
Asking for help	NaiveBayes	78.79	0.784	0.788
	KNN	89.79	0.9	0.898
	SVM	91.36	0.915	0.914
Verbal Sexual Advances	NaiveBayes	80.32	0.843	0.801
	KNN	84.15	0.854	0.842
	SVM	90.154	0.902	0.91
Questions	NaiveBayes	72.48	0.758	0.725
	KNN	89.6	0.897	0.893
	SVM	88.68	0.89	0.88

Table 5.6: Evaluation with combined features with automatic transcription

bag-of-word features with acoustic features.

5.2.3 Detecting Cursing

As shown in Table 5.7 we evaluate cursing detection using only acoustic features (from section 5.1.1), our cursing detection approach (shown in section 5.1.2) and a combination of both, where output of our cursing detection approach is used as a binary feature. We have used the SVM classifier as a detection classifier for this evaluation. As shown in Table 5.7, cursing detection using only acoustic features results in a low precision and recall rate of 75.1% and 77.4%, respectively. When manual transcription data is used for evaluation, our cursing detection approach detects all of the 91 single sense curse words and 41 of the multiple sense curse words. As shown in table 5.7 the precision rate for the multiple sense curse word detection is 87.23% and the recall rate is 89.13% and for overall curse detection the precision rate is 95.6% and the recall rate is 96.35%. The multiple sense curse words have word senses that varied from 2 to 9. If we try to detect the specific word sense for the curse words with multiples senses, the detection evaluation, precision rate goes down to 72.7% which shows that our binary word sense adaptation of Adapted Lesk algorithm [17] improves the curse word detection performance.

Since, many of the curse words are complex and uncommon in general English vocabularies,

Transcription	Features	Precision rate	Recall rate
	Acoustic	75.1	77.4
Manual	Textual	95.6	96.35
	Acoustic and textual	95.6	96.35
Automatic	Textual	93.9	91.2
	Acoustic and Textual	93.9	91.2

Table 5.7: Evaluation of cursing detection

the transcription error rate for software like Dragon is higher for them. After a short training of curse words transcription accuracy improves significantly. After short training, with automatic transcription by Dragon we achieved 93.9% precision and 91.2% recall for overall curse detection using our cursing detection approach (shown in section 5.1.2).

According to our evaluation as shown in Table 5.7, combining acoustic inference from speech with our cursing detection approach from transcribed speech does not improve accuracy. Hence, we conclude that our cursing detection approach (as shown in section 5.1.2) is sufficient to detect cursing from speech.

5.2.4 Detecting Repetitive Sentences

A study [204] on agitated demented elderly patients has shown that $50-80\%$ of them suffer from palilalia, which is a speech disorder characterized by the involuntary repetition of syllables, words, or phrases. Hence, it is highly unlikely of them will repeat large sentences with many words skipped. We have evaluated our algorithm on 80 speech samples collected in controlled experiments. Each of the converted text of these speech samples contains sentence repetition with at most 3 words skipped. Using manual and automatic transcription detection, the accuracy was 100% and 98.7%, respectively.

5.3 Real Patient Evaluation

We have also evaluated verbal event detections approaches on real agitated dementia patients using audio clips collected in realistic settings from elderly suffering from dementia. The clips (N=107) were collected for an NIH-funded randomized clinical trial (ClinicalTrials.gov Identifier: NCT01324219) that tested whether improved nursing home staff communication reduces challenging behaviors in persons with Alzheimer's disease and other dementias [205]. The clips were collected during morning care activities in 16 midwestern nursing homes. Duration of the audio clips vary between one to 30 minutes. The clips contain examples of agitated verbal events as well as periods without agitated verbal events. In total, 34 residents were included in the clips ranging in age from

63 to 98 years old (Mean=88, Standard Deviation = 7.2) and were 70% female, 97% Caucasian non-Hispanics, and 67% were prescribed psychotropic medications. Table 5.8 shows the number of agitated verbal events that appeared in those audio clips. The distribution of verbal events shows that verbal event questions are more common in agitated elderly suffering from dementia. There were no examples of verbal sexual advances. This data is representative of the fact that these events are not common for agitated elderly in nursing homes suffering from dementia.

Verbal events	Total (136 events in 107 clips)
Asking for help	11
Verbal sexual advances	0
Questions	52
Cursing	10
Repetitive sentences	14

Table 5.8: Verbal events from real patients.

Speech from agitated elderly suffering from dementia may vary from generic cases, but we have a relatively small dataset of 107 audio clips from 34 read agitated dementia patients. Hence, we apply the solutions from section 5.2 where we performed a leave one out cross validation on real patient data, additionally incorporating our collected data (as shown in section 5.2.1) from previous steps in the training set. Note that audio clips from real dementia patients contain extensive amount of hiss, hum, and other steady noises as well as music, coughing, door and window movements, beep sounds from air conditioners, etc. We used a spectral noise gating algorithm to reduce steady noises like hiss or hum sounds. To eliminate the effect of environmental noise, we have included environmental sounds (music, coughing, door and window movements, beep sounds, etc.) in the training set. For example, to evaluate the performance of asking for help detection using a binary classifier with one clip collected from the real patient data being tested, we included all the data from real patients and our other collected data (as shown in section 5.2.1) in the training set. Also, examples for environmental noise are included as negative examples for this binary classifier in the training set. The evaluation results of verbal event detection using real dementia patient data with manual transcription is shown in Figure 5.8 where performance metrics for verbal event detection remain approximately similar to our evaluation with our collected data (as shown in section 5.2.1). Figure 5.9 shows the evaluation of verbal events with automatic transcription on real patient data. Due to presence of noise, transcription error rate for real patient audio data was higher compare to our previous evaluations, which reduce the event detection accuracy. In this evaluation the SVM classifier is used as a detection classifier.

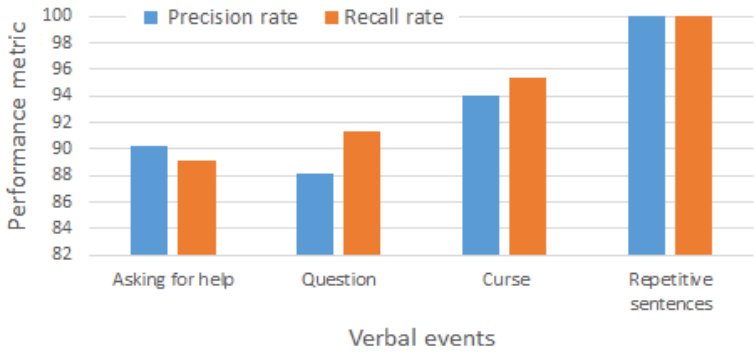


Figure 5.8: Real patients evaluation with manual transcription

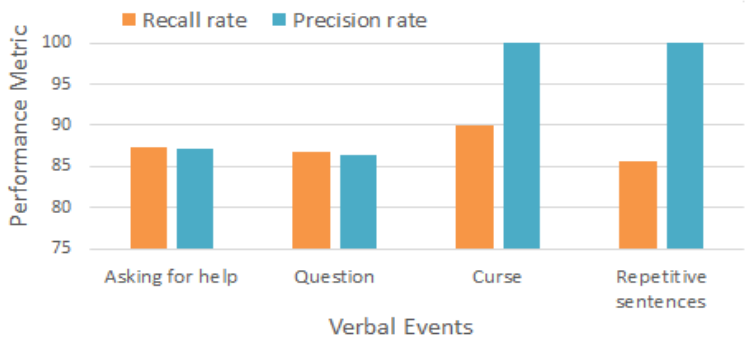


Figure 5.9: Real patients evaluation with automatic transcription

5.4 Discussion

A major challenge that we solved was detecting verbal agitation for dementia patients who mumble, speak in low volume, and don't articulate words very well. The value of this detection is clear from the medical community which uses them in their Cohen-Mansfield metrics to help in treatment. Significantly, we also showed that our solutions generalizes to the healthy population. The value for the healthy population is less obvious. However, applications of our solution include online video sharing sites such as *Youtube* and movies, where providers and users are able to detect objectionable content such as cursing, sexual advances, etc. to impose restrictions (e.g., for children). Detection of asking for help and questions can improve several human computer interaction (HCI) systems such as: automated customer service interaction systems, smart classrooms, etc. Also, some of the vocal events such as: asking for help, verbal sexual advances, and cursing are important for home safety.

While the focus of this chapter is to provide the details of the algorithmic solutions and their

evaluation, it is possible to incorporate the solutions into a working system. In fact, we have implemented the solutions on a Kinect system and deployed it in three homes with healthy people. Figure 5.10 shows the evaluation of the home deployments using automatic transcription. According to this evaluation, the performance of the classifiers (SVM classifiers) for all the verbal events are approximately similar to the evaluation using our collected data (as shown in section 5.2.1). Using this system with dementia patients will require a multi-year pilot study which is beyond the scope of this study.

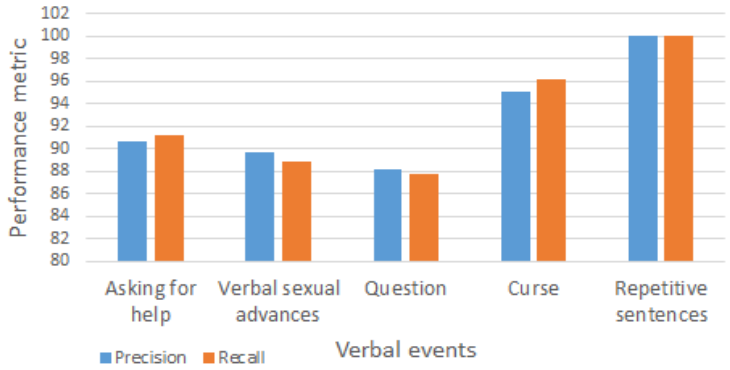


Figure 5.10: Detection of verbal events in homes

DAVE also addresses one key aspect of privacy. It keeps the recorded acoustic data private and only presents the type and time of occurrences of agitated vocal events through its interface. For example, a graphical representation of the change of frequency of agitated behavior can be displayed and then used by the caregiver to help diagnose the state of the disease of a patient.

5.5 Summary

This chapter discussed DAVE, a comprehensive set of event detection techniques to monitor and detect 5 important verbal behavioral events: asking for help, verbal sexual advances, questions, cursing, and talking with repetitive sentences. The novelty of DAVE includes combining acoustic signal processing with three different text mining paradigms to detect verbal events (asking for help, verbal sexual advances, and questions) which need both lexical content and acoustic variations to produce accurate results. To detect cursing and talking with repetitive sentences we extend word sense disambiguation and sequential pattern mining algorithms.

The solutions [162, 163] have applicability to monitoring dementia patients, for online video sharing applications, human computer interaction (HCI) systems, home safety, and other health care applications.

We have provided an extensive evaluation that includes audio clips collected from real agitated

elderly patients suffering from dementia, *Youtube*, movies, online data repositories, controlled experiments, and home deployments.

Chapter 6

Ambient Human Events

The purpose of this chapter is to present an ambient human event detection (AHED) platform able to analyze, identify, and detect specific unusual or deviant human events. Such a system is especially useful for patients who have a reasonable degree of both physical and mental autonomy to carry on regular Activities of Daily Living (ADL), but, because of their age or disease, need to be continuously monitored. Ambient human event detection (AHED) platform has application in home hygiene and health monitoring (i.e., brushing teeth, snoring), child-care (i.e., baby cry detection), monitoring for public safety and security (i.e., gunshot,cry, scream), and on urban noise assessment of residential area.

In recent years smart technologies such as smart homes, smart cars, home health monitoring and surveillance systems, have become popular among consumers. Smart speakers, such as, alexa, google home, come with built in microphones. In most modern cars a microphone already exists in the cabin. Hence, a real-time AED system, capable of running locally on resource constrained devices, such as, Raspberry Pi, can be a great real-time monitoring solution, and can be added to already existing smart home and smart car devices.

Though there are some available datasets [137, 52, 138] which contain event level annotation for automated ambient human event monitoring systems, the amount of labeled event data is very small. Hence, the majority of AHED studies [190, 74] perform their evaluation on small datasets. A limited dataset in training leads to lack of robustness of the AHED approach as they are used in different environments (with noise and a large variety of extraneous sound events).

This chapter presents a novel framework for AHED, which generates robust models for audio monitoring applications with limited available data. Moreover, the generated AHED systems are real-time executable on resource constrained devices. The main characteristics and contributions of the framework and its evaluation are:

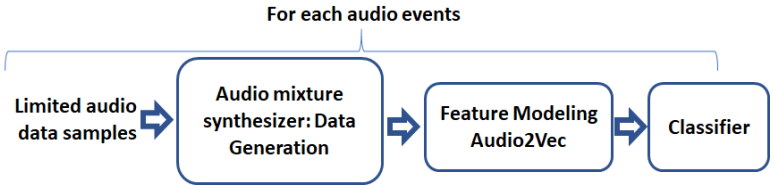


Figure 6.1: Framework for real-time AHED with limited data.

- As shown in figure 6.1, for each of the ambient audio events with limited data, the framework generates a large synthetic dataset with a large variation of background environmental sounds, signal to noise ratios (SNRs), and reverberation effects. Theoretically, the presented **automated audio mixture synthesizer** (section 6.1) can generate an infinite number of variations.

- To extract meaningful and effective feature representation from the raw audio data, this chapter presents a novel **computationally effective feature modeling/engineering technique, named Audio2Vec** (section 6.2.1 & 6.2.2). The generated representations by Audio2Vec are robust against environmental noise, reverberation, and de-amplification of sound due to distance. Moreover, it identifies and exploits the inherent relation between audio states and targeted audio events. As a result, Audio2Vec features can be used with much shallower (less layers & network parameters) neural network classifiers, and achieves significantly higher accuracy compared to the baseline feature representations typically used with much deeper neural networks. Also, shallow networks (for classification) have less execution time which makes them more suitable for real-time AHED systems on resource constrained devices.

- To demonstrate the extensive applicability of the presented AHED framework, we applied the framework (figure 6.1) to develop and evaluate ambient audio detection models for ten ambient human audio events: crying, laughter, screaming, coughing, snoring, brushing teeth, sneezing, baby crying, glass breaking, and gunshots. One example of the value of detecting multiple audio events is that according to the Cohen-Mansfield Agitation Inventory [35], detection of crying, laughter, and screaming are important for monitoring agitation in dementia patients. Also automated detection of coughing, snoring, brushing teeth, sneezing have application in home healthcare and hygiene monitoring. Detection of baby crying is important for infant monitoring systems, and gunshots and glass breaking have application in automated surveillance and security systems. Our AHED approach using the Audio2Vec feature representation achieved on average 10.3% higher F_1 score compared to the best baseline approach for 10 targeted ambient human audio events (section 6.3.2).

- To evaluate the applicability of our approach in realistic scenarios, running on resource constrained devices (on-device or local computations), we implemented an **real-time AHED**

system (that simultaneously detects the 10 targeted audio events) on a Raspberry Pi 3B with a MATRIX Creator development board (section 6.3.3). We evaluated the implemented system for two realistic applications: real homes and inside car monitoring. According to the evaluation we achieved average F_1 scores of 0.96 and 0.956 for AHED in real-home and in-car settings, respectively.

- An effective ambient human event monitoring system needs to be real-time executable. We experimentally evaluated the CPU run-time of each component of our AHED system and demonstrated its real-time capability for a constrained device: Raspberry Pi (section 6.3.4).

6.1 Synthetic Dataset

A robust ambient human audio event monitoring system needs models that perform well in various environments not introduced in the training phase. Also, in real event monitoring scenarios input audio signal to noise ratio (SNR) can be very low due to variable source to microphone distances and presence of other ambient noise sources. Additionally, in indoor settings, audio data suffers from reverberation effects. Training a supervised model robust against unknown environments, reverberation, and low SNR requires sufficiently large dataset with variation of environmental sounds, reverberation, and SNRs. One of the challenges this chapter addresses is having small available audio event datasets. But there are larger environmental scene datasets, where, background sound is collected from many different environments, such as, cafes, train stations, or parks. These sounds are easy to collect and do not need any labeling of the data. This chapter presents an audio mixture synthesizer, that generates a large synthetic mixture of labeled isolated audio event clips and various environmental audio clips. Using this automated generalized approach, it is possible to generate any number of well labeled positive and negative synthetic data samples (this can be applied to any audio event with a small available dataset).

Solution: Audio mixture synthesizer: We mix isolated audio clips (from available small datasets) with environmental background sounds to generate synthetic data samples. For the mixture synthesizer we used the pydub python toolkit. Synthetic audio samples are generated in the following manner.

We randomly select 10s audio from a randomly selected environmental background audio clip. Additionally, we randomly select one or more isolated (targeted and/or non-targeted) audio event clips. Both the environmental background audio and the isolated event audio clips are amplified or de-amplified to generate a random event-to-background ratios (EBR) between -6 to 6 dB. Then isolated audio event clips are overlaid on the 10s background audio clips at randomly selected positions. Random numbers are drawn from a uniform distribution, to achieve maximum variation in background sounds, EBR, and event positions.

In a 10s synthetic positive audio sample of a targeted audio event (e.g., screaming), at least one

(or more) isolated targeted audio clips are overlaid/placed, and zero or more non-targeted audio clips are overlaid/placed on the same 10s environmental sound. In a 10s synthetic negative sample of a targeted audio event, any other (except the targeted event) or none of the isolated audio clips are overlaid/placed in the 10s audio.

There are different artificial reverberation effect parameters to model how sound waves reflect from various types of room size and characteristics. Our synthesizer introduces different combinations of reverberation parameters: wet/dry ratio, diffusion, and decay factor in the generated audio samples. These parameters generate different reverberated signal to the original signal ratio, discrete echo effects, and reverberation tails (decay factor of reverberation).

The generated synthetic data is highly imbalanced (due to significantly larger number of non-targeted audio clips), that can make the binary AHED classifiers biased toward the majority (i.e., negative) class. To address the data imbalance issue, we perform cluster-based under-sampling [210] on the negative (non-targeted) 10s audio clips to generate an equal number of positive and negative synthetic data samples for each of the targeted audio events, as well as to accommodate the variations of the negative or non-targeted audio samples.

6.2 Audio Event Detection

Our presented approach performs the AHED task on 10s audio clips. We developed a novel feature modeling approach named Audio2Vec (section 6.2.2) to extract robust and effective feature representations of audio data. Binary classifiers for each of the targeted events (section 6.2.3) take the Audio2Vec features as input and perform the classification task.

6.2.1 Audio Features

Our approach segments the generated 10s audio clips into overlapping windows (200ms with 20% overlap in our evaluation), and extracts a feature set from each window. The extracted feature set, represents the inherent state of audio from that window. Based on the previous studies on acoustic features associated with targeted audio events (i.e., baby crying, gunshots, glass breaking, screaming) we considered low-level descriptor (LLD) features shown in left column of Table 6.2.1, as well as their delta and delta-delta coefficients. Each window is segmented into overlapping 25ms frames with 10 ms overlap, from which LLD features are extracted. Next the 8 functionals: minimum, maximum, mean, median, standard deviation, variance, skew and kurtosis, are applied to extract the audio window representation. In total 272 raw features are extracted from each of the 200ms windows, where 10s audio clips consist of 62 overlapping windows (200ms).

LLD Features	High level features (functionals)
Zero crossing rate & Δ (2-dim)	Min, Max
Energy & Δ (2-dim)	std, skew, var
Spectral centroid & Δ (2-dim)	kurtosis, mean
Pitch & Δ (2-dim)	median
MFCC & Δ (26-dim)	

Table 6.1: Raw audio features

6.2.2 Feature Modeling

The modeling stage of an audio analysis system develops a representation that reflects the audio information for that specific task. Each segment of audio represents a state, and an ambient human audio event is represented by the progression of audio signals through various states. The following sections introduce a robust novel representation of the audio state from a 200ms window that takes into account the inherent notion of that state with a particular targeted audio event.

Audio to Word

We use the Audio-Codebook model [150] to represent the audio signal from windows with 'audio words'. The 'audio words' represent the state of audio in each 200ms window. In our context the audio-codebook words are fragments of audio signal represented by features. We need a robust feature descriptors to represent the audio state in an audio window. Inspired by [85], we use a GMM based clustering method to generate the audio-codebook from the functional representations mentioned in section 6.2.1.

In the codebook generation step, a GMM based model is trained on randomly sampled data from the training set. The resulting clusters form our codebook audio words. Once the codebook has been generated, acoustic features within a certain range of the audio signal are assigned to the closest audio words (cluster centers) in the codebook. In the experiments, we have evaluated with different sizes of codebooks.

The raw audio features from section 6.2.1 distort up to a certain level with variance of environmental noise, audio de-amplification, and reverberation. Our trained audio-codebook places similar points in the feature space into the same code words, which make the feature representation robust against different environments, noise, and distance to the microphone.

Audio2Vec Approach

Audio words extracted from overlapping 200ms windows represent the states of the audio. Since, audio signals from a particular audio event are different from others, audio states representing that event would be different from others. Also, some states occur more frequently in a targeted audio event signal compared to other events. Typical audio-codebook word methods fails to convey this

state to an audio event representation.

To identify and exploit the inherent relation between audio states and audio events, we developed a novel audio word to vector conversion (Audio2vec) approach, that generates an N dimensional vector representation for each of the audio words (from the Audio-codebook). Position of a generated vector in the N dimensional vector space, depicts the relation between the state it represents and the targeted audio event.

Algorithm 1 shows our Audio2Vec conversion approach. In the initialization stage audio words unique for a targeted audio event are randomly assigned vectors near the $Positive_{centre}$ vector in an N dimensional vector space and words which never occur in the targeted events are assigned random vectors near $Negative_{centre}$. Other audio words are randomly assigned vectors between them ($Positive_{centre}$ and $Negative_{centre}$). In the iteration stage, every time an audio word w_j occurs in the targeted event, the vector representation of that audio word v_j is modified according to equation 6.1, which makes v_j move closer to $Positive_{centre}$ in the N dimensional vector space. Otherwise v_j is modified according to equation 6.2, which makes v_j move closer to $Negative_{centre}$ (line 18-24, Algorithm 1).

$$v_j \leftarrow v_j + (Positive_{centre} - v_j) \times \delta_p \tag{6.1}$$

$$v_j \leftarrow v_j + (Negative_{centre} - v_j) \times \delta_n \tag{6.2}$$

Since, the targeted audio events are only few seconds in the 10s audio samples, the total number of audio words that appear in the targeted audio event segments are significantly lower compare to total number of audio words in other audio event segments. To mitigate this bias, we calculate the addition fraction parameter for negative samples δ_n according to equation 6.3, where N_p and N_n are the total number of audio words in the targeted and other events, respectively.

$$\delta_n \leftarrow \frac{N_p}{N_n} \times \delta_p \tag{6.3}$$

Figure 6.2 shows an example of the Audio2Vec approach, where the vector dimension is N=2. In Figure 6.2 (a), black points are the vectors (audio words) unique for targeted events, white points are ones that never occur in the targeted events, and the grey ones are common between two classes. Later, in the iterative stage of Audio2vec, every time an audio word occurs in the targeted event in training set, the vector representation of that audio word is moved closer to the targeted event clusters in the vector space, as shown in Figure 6.2 (b). Similarly, if an audio word occurs for any other events, the vector representation of that audio word is moved further from the targeted event clusters (figure 6.2 (c)). As shown in Figure 6.2 (d), the Audio2vec approach brings frequently occurring words in the targeted events closer in the vector space compared to others. The advantages of the Audio2Vec approach:

Algorithm 1 Audio2Vec Algorithm

w: audio word
v: Audio2Vec vector
C: audio-codebook
A: audio clip represented as a sequence of n audio words; $A = \{w_1, w_2, \ldots, w_n\}$
T: set of audio clips for training
δ_p: small constant used as parameter
δ_n: small constant used as parameter
m: dimension of generated Audio2Vec vector
$iter$: number of iterations

procedure AUDIO2VEC ALGORITHM(T, C, δ_p, δ_n, $iter$)
    #Initialization:
    $Positive_{centre} \leftarrow [k_1 k_2 \ldots k_m]$ where $k \in \{0.8, 1\}$
    $Negative_{centre} \leftarrow [k_1 k_2 \ldots k_m]$ where $k \in \{0, 0.2\}$
    for all audio word $w_i \in C$ **do**
        if w_i only appears in targeted audio event **then**
            $v_i \leftarrow [k_1 k_2 \ldots k_m]$ where $k \in \{0.8, 1\}$
        end if
        if w_i never appears in targeted audio event **then**
            $v_i \leftarrow [k_1 k_2 \ldots k_m]$ where $k \in \{0, 0.2\}$
        end if
        if w_i appears in targeted and other events **then**
            $v_i \leftarrow [k_1 k_2 \ldots k_m]$ where $k \in \{0.4, 0.6\}$
        end if
    end for
    loop: $iter$ times
        for all audio clip $A_i \in T$ **do**
            for audio word $w_j \in A_i$ **do**
                if $w_j \in$ targeted audio event **then**
                $difference \leftarrow Positive_{centre} - v_j$
                $v_j \leftarrow v_j + difference \times \delta_p$
                else
                $difference \leftarrow Negative_{centre} - v_j$
                $v_j \leftarrow v_j + difference \times \delta_n$
                end if
            end for
        end for
    end loop
end procedure

Representational efficiency: Audio2Vec learns to map audio data from each 200ms windows into a fixed-length low-dimensional continuous vector space from their distributional properties observed in training. Our evaluation has found that the best Audio2Vec dimension is N=30. Hence, Audio2Vec approach represents the audio windows by a significantly low-dimensional distributed representation. Classifiers that take lower dimensional input data can optimize their parameters more effectively when training data is limited.

Mapping Efficiency: An interesting property of Audio2Vec vectors is that they encode the syntactic relationships between audio states and targeted audio events (classes). Audio2Vec vectors are similar for audio states with similar probability of occurring in targeted audio events. These characteristics are similar to the output of convolution layers in a CNN. Each convolution layer generates a successively higher level abstraction of the input data, which preserves essential yet unique information. Deep CNNs extract meaningful feature representations (also in the form of vectors) from input data by employing a deep hierarchy of layers. For example, the best baseline CNN classifier (section 6.3.2) has 4 convolution layers, with, in total, 140501 network parameters. Training such a high number of parameters needs a large training set. One of the challenges of this study is having a small amount of training data. The advantage of Audio2Vec approach is, its vector generation process involves vector addition and subtractions observing the training examples and is not constrained by number of training instances. Hence, the Audio2Vec approach can generate effective feature representations for our targeted application with limited training examples.

Execution efficiency: The Audio2Vec approach performs the clustering operation (to calculate Audio-Codebook) during training and stores the audio-words to Audio2Vec vectors conversion maps in a dictionary. Hence, during execution, converting raw audio from 200ms windows to Audio2Vec vectors involves finding the nearest cluster centroid and a dictionary lookup, which are linear in complexity and require significantly fewer computations compared to the baseline deep learning approaches.

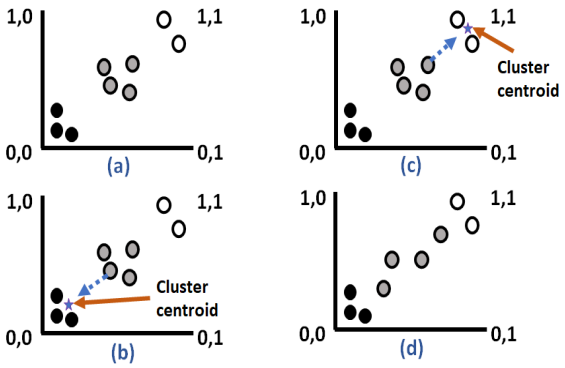


Figure 6.2: Example of Audio2vec approach in 2 dimensional space.

6.2.3 Classifier

In this study we evaluated with Convolutional Neural Network (CNN), Bi-directional Long Short-Term Memory (BLSTM), and Deep Neural Network (DNN) (i.e., 'vanilla' Neural Network) as classifiers. For each of these classifiers we performed a grid search on the network parameter values. For each of the classifiers, our evaluation iterate on 1 to 10 layers, with 50 to 500 neurons/filters (for convolution layers) for each of the layers. Due to the limitation of space, each of the classifiers with only the best iterated parameter combinations on the training set is presented in the evaluation section.

6.3 Evaluation

Our evaluation consists of three parts. First, section 6.3.1 & 6.3.2 discuss our synthetic dataset generation for 10 targeted audio events, and the performance of our Audio2Vec AHED approach on the generated data. Later in section 6.3.3, we evaluate our AHED approach for two realistic applications: in-home and in-car audio monitoring systems. And, in section 6.3.4, we evaluate the CPU run-time of each component of our system to demonstrate it's real-time capability.

6.3.1 Synthetic Data Generation

Our presented system detects ten audio events. We collected 160 isolated audio clips for each of these targeted events (100 for training, 30 for testing, and 30 for validation) from the freesound dataset [53], the ESC environmental sound Dataset [138], and the MIVIA audio event dataset [52].

To train a robust audio detection classifier, we need a training dataset that contains a large variation of non-targeted events that may occur in the real scenarios. Hence, we collected isolated audio clips of 80 environmental and human events (40 clips for each category) from the freesound dataset [53] and the ESC Dataset [138]. These events include environmental events such as, rain, sea waves, birds, water drops, wind, pouring water, car horn, helicopter, siren, engine, train, bells, fireworks, and human sounds such as, clapping, breathing, footsteps, drinking, sipping, dish washing, and animal sounds such as a dog, rooster, cow, cat, insects, crow, etc. Additionally, we collected 400 clips of human speech with different emotions (happy, angry, sad) from the EMA speech dataset [93]. We use this large variation of non-targeted isolated audio clips to generate negative training samples.

Collected isolated sound clips (targeted and non-targeted) have exact labels with sampling frequency 44.1 kHz or higher. The duration of the audio clips varied from 0.5 to 4.3 seconds.

To introduce a large variety of environmental background sounds we collected 1121 background audio clips from the TUT Acoustic Scenes development dataset [110]. This environmental scene dataset contains 15 acoustic scenes, including audio clips recorded from bus, train, cafe, car, city center, forest, store, home, beach, metro stations, office, park.

Using our mixture synthesizer (section 6.1), we generate 2000, 10s audio clips (1000 for training, 500 for validation and 500 for testing), for each of the targeted audio events. Isolated audio event clips and environmental background audio data for training, validation, and testing datasets were disjoint. We perform 10-fold cross-validations on the training and testing sets to select the best models and model parameters that fit the data. We use the validation set to report the AHED performance on the synthetic dataset.

Due to the highly imbalanced data in the evaluation phase of the study, using accuracy as the evaluation metrics can introduce an accuracy paradox. Hence, we used class-wise F_1 score, the harmonic average of precision, and recall as the evaluation metrics.

6.3.2 Evaluation Results: synthetic data

In this section, we describe the binary audio event detection evaluation results on generated synthetic data (section 6.3.1). As mentioned in section 6.2.1, each 10s audio clips is segmented into 62, 200ms windows and 272-dimensional feature-set is extracted from each of these windows. Our Audio2Vec feature modeling approach converts these 272 dimensional raw feature-sets to N dimensional vectors. As the first step of Audio2Vec approach, we generate 1000 GMM clusters from randomly selected audio samples for each of the targeted audio events. The resulting clusters form the generated audio words (section 6.2.2). The proposed Audio2Vec solution generates 30-dimensional vector representation for each of the extracted audio words as described in Algorithm 1. We performed a grid search over 500 to 3000 cluster sizes and 10 to 100 dimensions to find the best values of cluster size (1000) and Audio2Vec vector dimension (N=30) for our AHED task.

Figure 6.3 illustrates the change of generated Audio2Vec feature vector space, for the audio event: gunshot with the increase of iteration number of Algorithm 1. PCA based visualization approach is used here to visualize the 30-dimensional feature space in 2-dimensions. In this figure, the blue points represent the audio words unique for gunshot signals, red points represent the ones that never occur in gunshot signals, and green points represent the ones which are common. At the initial step of Algorithm 1, green points are clustered in the middle between red and blue points (Figure 6.3(a)). With the increase of iterations, these points spread out, and as points occur more frequently in gunshot signals they move toward blue points' cluster and the points that occur frequently in other audio signals move towards the red points' cluster. This visualization demonstrates how Audio2vec approach brings frequently occurring audio words in the targeted event (gunshot) closer in the vector space compared to others.

Figure 6.4 shows the evaluation (class-wise F_1 scores) on different deep learning classifiers for our AHED task. In this evaluation all three classifiers take extracted 62×30 dimensional Audio2Vec features (from 10s audio) as input. We performed a grid search on the network parameters to identify the best combination. The CNN implementation had 2 convolution layers [60,60], each with 60 convolution kernels (temporal extension of each filter 2), a ReLU activation function, 20% dropout

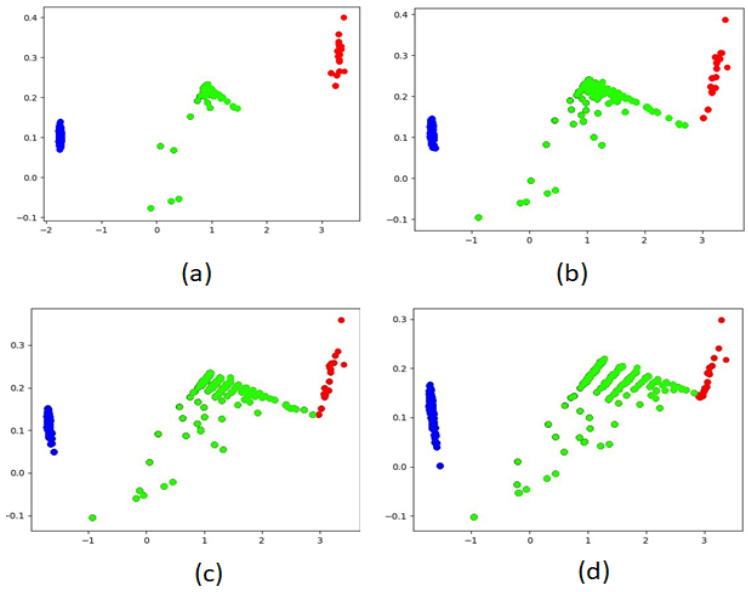


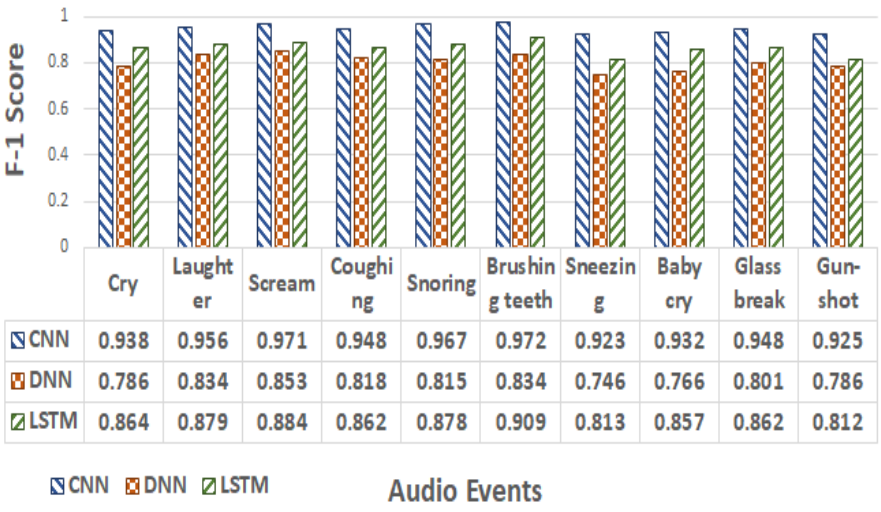
Figure 6.3: Iterations of Audio2Vec vector generation approach for gunshots. Figure (a), (b), (c), and (d) show the generated vector feature space after initial, 5,10 and 20 iterations of Algorithm 1.

rate and max pooling (window size 2 and down-scaling factor 2). Two fully connected dense layers [20,1] with sigmoid activation function, were attached, which make binary event presence decision. The CNN classifier achieved an average F_1 score of 0.948 for the ten targeted events.

The BLSTM implementation had two layers with [100, 100] nodes, a 20% dropout rate, and two fully connected dense layers [20, 1] with sigmoid activation function. The BLSTM classifier achieved an average F_1 score of 0.862 for the ten targeted events. DNN implementation had four fully connected layers with [500, 500, 300, 100] nodes and a ReLU activation function, a 20% dropout rate, and one fully connected dense layer [1] with a sigmoid activation function to make binary decisions. The DNN classifier achieved an average F_1 score of 0.8039 for the ten targeted events.

Our targeted audio event duration varied between 0.5s to 4.3s. Hence, the BLSTMs ability to convey contextual information in long audio sequence was not very advantageous. In CNNs, the convolutional filters not only can generate meaningful feature representations, moreover, they are translation invariant. That means, during training the convolution filters are being applied in a sliding window fashion on the entire 10s audio. Hence, no matter where an audio event occurs in a 10s audio clip, CNNs can detect it with limited training examples.

The CNN classifier achieved 9.9% and 17.9%, higher F_1 scores compared to the BLSTM and DNN classifiers. According to this evaluation we achieved the highest F_1 scores of 0.971 and 0.972, for events screaming and brushing teeth. The detection of gunshots and sneezing were most difficult since, different environmental sounds (from TUT Acoustic Scenes) are very similar to them,



	Cry	Laughter	Scream	Coughing	Snoring	Brushing teeth	Sneezing	Baby cry	Glass break	Gun-shot
CNN	0.938	0.956	0.971	0.948	0.967	0.972	0.923	0.932	0.948	0.925
DNN	0.786	0.834	0.853	0.818	0.815	0.834	0.746	0.766	0.801	0.786
LSTM	0.864	0.879	0.884	0.862	0.878	0.909	0.813	0.857	0.862	0.812

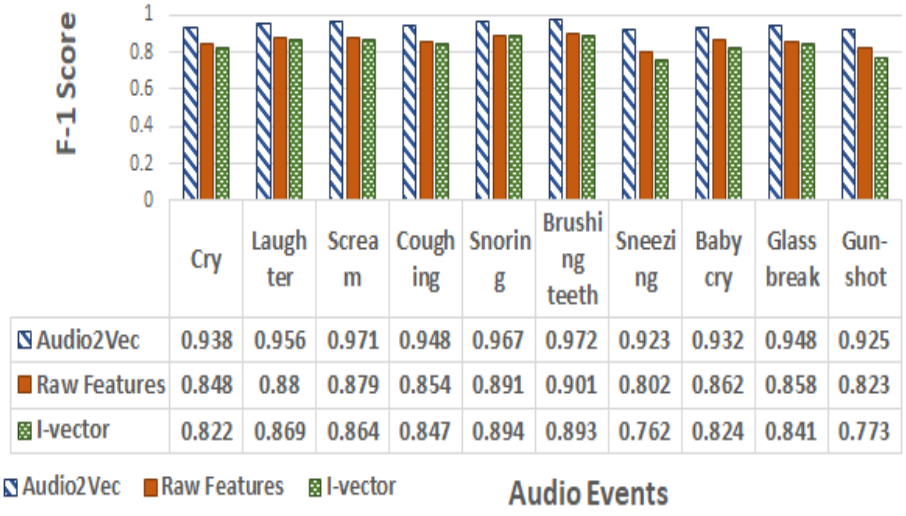
Figure 6.4: Evaluation of classifiers with Audio2Vec features.

especially after the de-amplification and reverberation effects.

To evaluate the effectiveness of Audio2Vec representation, we analyze the performance of AHED with three different feature representations: 1) Audio2Vec, 2) I-vector, and 3) raw acoustic feature-set discussed in Section 6.2.1. We performed a grid search on the three classifiers (CNN, BLSTM, and DNN) parameters to identify the best classifiers. As shown in figure 6.5, raw feature-set with a CNN implementation achieved an average F_1 score of 0.8598. The CNN implementation had 4 convolution layers, each with 100 convolution kernels (temporal extension of each filter 4), a ReLU activation function, a 20% dropout rate and max pooling (window size 2 and down-scaling factor 2). Two fully connected dense layers [100,1] with a sigmoid activation function, are attached, which makes the binary event presence decision.

The I-vector system is a technique to map the high-dimensional GMM super vector space to low-dimensional space called total variability space. The basic idea of using I-vector representation is to represent each 200ms windows using concatenated I-vector feature vectors extracted based on audio event-specific GMM super vectors, and then to use these in the classifier. However, the existence of noise and channel variation can substantially affect the performance of i-vector representations. Since, environmental background sounds for training and validation data in our generated synthetic dataset are disjoint and the EBR varied between -6 to 6 dB, the i-vector representation fails to achieve a better AHED performance. I-vector features with a CNN implementation achieved an average F_1 score of 0.839.

According to this evaluation the AHED with our Audio2Vec features achieves 10.3% higher F_1 score compared to the best baseline features.



	Cry	Laughter	Scream	Coughing	Snoring	Brushing teeth	Sneezing	Baby cry	Glass break	Gun-shot
Audio2Vec	0.938	0.956	0.971	0.948	0.967	0.972	0.923	0.932	0.948	0.925
Raw Features	0.848	0.88	0.879	0.854	0.891	0.901	0.802	0.862	0.858	0.823
I-vector	0.822	0.869	0.864	0.847	0.894	0.893	0.762	0.824	0.841	0.773

Figure 6.5: Evaluation on features.

6.3.3 Evaluation on Realistic Applications

The MATRIX Creator [104] is an all-inclusive development board that connects to the GPIO pins on the Raspberry Pi. It has an 8-microphone MEMS array and an ARM Cortex M3 microcontroller and features built-in noise cancellation and beamforming. We used the MATRIX Creator with Raspberry Pi 3B, as our constrained AHED device (shown in figure 6.6-A).

We evaluate our AHED approach for two realistic applications: inside car and real home audio event monitoring. For home event monitoring evaluation, we collected audio data from a pseudo smart home (figure 6.6-B,C) setup in the Smart Home Lab at the University of Virginia, and from a real one-bedroom apartment. In both settings we placed the AHED device in center of the room. Since performing some events in real home or car settings were not feasible (such as for gunshots), we played sounds of targeted events through a Sony SRS-XB10 Bluetooth speaker. The speaker was placed in different places of the bedroom, bath, kitchen and living room. We collected data for a single day where a single occupant performed daily in-home activities, and different events were played at random times from random places. In total, we collected 50 audio examples for each of the 10 targeted audio events from each of the home settings.

Figure 6.7 shows our evaluation on real home data. In this evaluation the AHED approach achieved an average F_1 score of 0.96 for the ten targeted events, that is 1.2% higher compared to our evaluation on the synthetic dataset (section 6.3.2). This is due to the significantly less noise and variations of non-targeted audio events, compared to some challenging pseudo scenarios (section 6.3.1), such as, trains, cafes, city center. According to the evaluation gunshot detection was most difficult since, different in-home environmental sounds, such as, knocking on the door, jumping,

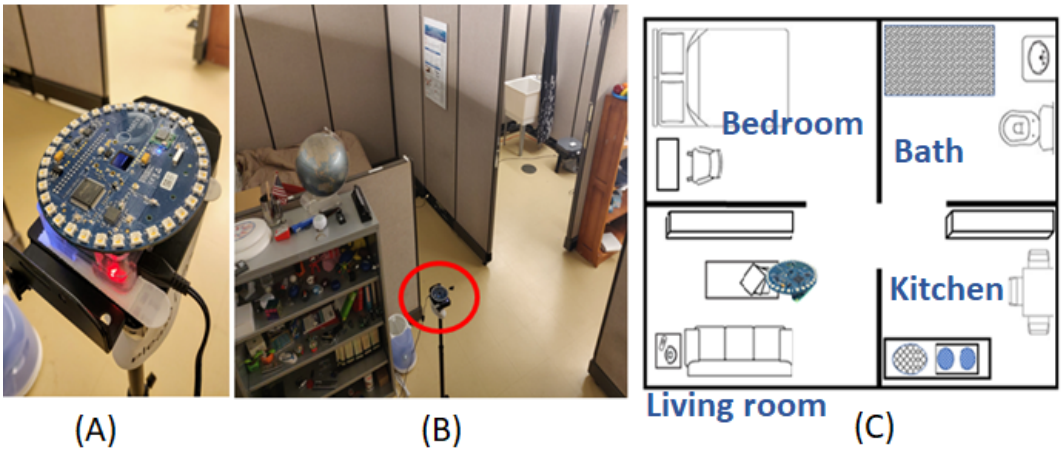


Figure 6.6: Realistic data collection.

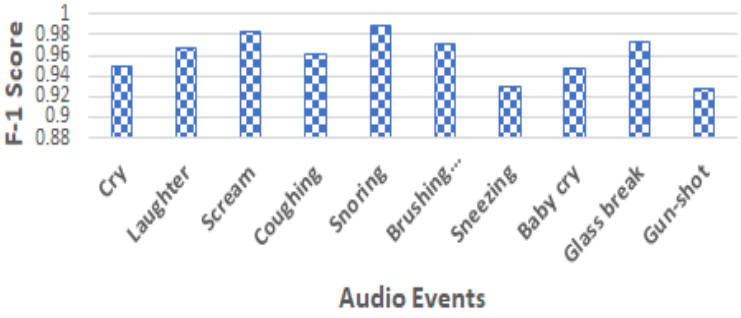


Figure 6.7: Evaluation in real home scenario.

walking on a wooden floor, are very similar to de-amplified gunshot sounds.

For the inside car event monitoring evaluation, we collected audio data from a Toyota Corolla 2016 car for 3 different speed ranges (below 25, 25 to 45, and above 45 MPH) with 2 different conditions, AC on and AC off. The AHED device was placed in the center of the car, and different sound events were played from a Sony speaker placed in four passenger seat positions. For each of the 6 conditions we collected 20 audio examples for each of the 10 targeted audio events from each passenger seat positions. Additionally, we collected audio samples without the presence of any of the targeted events.

All the audio examples of targeted and non-targeted events used for this evaluation (in-home & in-car) were not included in the synthetic dataset (section 6.3.1).

Table 6.2 shows our evaluation on real car. These results are comparable to our evaluation in section 6.3.2. At low car speeds, the AHED approach achieved an average F_1 score of 0.9685. That is due to the significantly less noise and the absence of de-amplification on event sounds (All passenger

Speed (MPH Condition)	0-25		25-45		45-65	
	AC on	AC off	AC on	AC off	AC on	AC off
Cry	0.954	0.946	0.954	0.948	0.94	0.938
Laughter	0.971	0.971	0.967	0.951	0.952	0.951
Scream	0.986	0.986	0.983	0.974	0.98	0.964
Coughing	0.968	0.968	0.964	0.961	0.948	0.942
Snoring	0.991	0.991	0.991	0.98	0.971	0.971
Brushing teeth	0.983	0.983	0.98	0.98	0.976	0.976
Sneezing	0.942	0.93	0.938	0.926	0.91	0.895
Baby cry	0.944	0.942	0.94	0.924	0.928	0.901
Glass break	0.989	0.982	0.971	0.971	0.961	0.942
Gunshot	0.957	0.957	0.942	0.928	0.901	0.896

Table 6.2: AHED evaluation in real car scenario.

seat positions are close to the AHED device). Though at high speeds, the AHED performance drops to 0.946 and 0.937 average F_1 scores on AC-on and AC-off conditions. The humming sound of AC, reduce the effect of noises in the car, hence, AC-on condition performed better for most of the cases.

6.3.4 CPU Time Benchmarking for Real Time execution

We did all our realistic application experiments on a Raspberry Pi 3B having a Quad Core 1.2GHz Broadcom BCM2837 64bit CPU and 1GB LPDDR2 (900 MHz) memory. Our program reads a 10s audio file at a time, and extracts 272-raw features. For each of the (10) targeted events a process reads the raw feature-set, converts it to an Audio2Vec representation, and performs classification through the CNN implementation described in section 6.3.2. We performed the multiprocessing tasks through Python multiprocessing package.

We benchmark the computation time for (1) extracting 272-dimensional raw feature-set from 10s audio, (2) combined time for 10 processes to convert raw features to respective Audio2Vec representations, and (3) combined time for 10 processes to perform classification, as shown in table 6.3. Moreover, we implemented the best baseline approach from section 6.3.2: Raw audio features with a 4-convolutional layer CNN implementation in the Raspberry Pi 3B, with the similar multiprocessing approach. As shown in table 6.3, for the baseline approach we benchmark the computation time for (1) extracting 272-dimensional raw feature-set, (2) combined time for 10 processes to perform classification taking raw features. Computation time is the time spent running the particular task plus running OS code on behalf of the task.

According to table 6.3, reading audio files and extracting raw feature-set takes high CPU execution times, since they involve I/O operations. Audio2Vec vector conversion takes only 0.022s.

Audio2Vec AHED approach				
Task	Reading audio and extracting raw features	Audio2Vec conversion (cumulative)	Classification (cumulative)	Total time
Time (sec)	5.32	0.012	0.215	5.547
Baseline: Raw features+CNN				
Task	Reading audio and extracting raw features	Classification (cumulative)		Total time
Time (sec)	5.32	0.338		5.658

Table 6.3: Computation time for various system tasks

The 2-convolutional layer CNN implementation used in our AHED approach has $38,561$ parameters, and cumulative classification time takes 0.215s. Hence, the cumulative time taken by the 10 processes is 0.237s. And end-to-end total AHED system time for one 10s audio is 4.557s. Given the AHED window is 10s, the real-time system is extendable to including many more audio events. The baseline CNN implementation has $150,601$ parameters, and cumulative classification time takes 0.338s. Hence, the cumulative time taken by the 10 processes (one for each events) in the baseline implementation is 42.6% higher compared to the Audio2Vec AHED approach.

According to this evaluation, our presented AHED approach is capable of real time execution on a resource constrained device. Note that CPU times reported in table 6.3 are when only the audio event detection program is running. Running additional programs will effect/change these times.

6.4 Summary

This chapter presents a novel framework [160] for robust ambient human event detection (AHED) models generation using limited available data. The framework uses a novel audio mixture synthesizer to generate a large synthetic dataset, that contains a large variation of background environmental sounds, noise, SNR, and reverberation effects; a novel robust and computationally effective feature representation technique, named, Audio2Vec. Due to the meaningful syntactic characteristics of the extracted feature representations, AED with Audio2Vec, performs significantly better with shallow network models, compared to much deeper models with baseline features. To demonstrate the applicability of the framework, we implemented a real-time AHED system in a Raspberry Pi 3B and evaluated its performance in real home and in-car settings, that achieved F_1 scores of 0.96 and 0.956, respectively. Moreover, we experimentally evaluated the CPU run-time of the AED system to demonstrate its on-device real-time capability for a constrained device. Our framework is extendable to any other audio events and the real-time AHED system is extendable to include many more ambient human audio events.

Chapter 7

Conclusion

As part of this thesis, we have developed a total of four systems. Each of which contributes to the primary objective of this dissertation which is to develop human verbal event monitoring systems addressing the open challenges in realistic health-care applications by leveraging novel and adaptive feature engineering approaches.

In this final chapter, we describe the key contributions it makes, its limitations, and provide directions for future improvements.

7.1 Summary and Key Contributions

This thesis makes the following key contributions:

7.1.1 Distant Emotion Recognition

Distant emotion recognition (DER) extends the application of speech emotion recognition to the very challenging situation that is determined by variable speaker to microphone distances. The performance of conventional emotion recognition systems degrades dramatically as soon as the microphone is moved away from the mouth of the speaker. This is due to a broad variety of effects such as background noise, feature distortion with distance, overlapping speech from other speakers, and reverberation. This thesis presents a novel solution for DER, addressing the key challenges by identification and deletion of features from consideration which are significantly distorted by distance (Distant feature selection, section 3.1), creating a novel, called Emo2vec, feature modeling/engineering and overlapping speech filtering technique, and the use of an LSTM classifier to capture the temporal dynamics of speech states found in emotions. A comprehensive evaluation is conducted on two acted datasets (with artificially generated distance effect) as well as on a new emotional dataset of 12 spontaneous family discussions (total 38 participants) with audio recorded

from multiple microphones placed in different distances. Our solution achieves an average 91.6%, 90.1% and 89.5% accuracy for emotion happy, angry and sad, respectively, across various distances which is more than a 16% increase on average in accuracy compared to the best baseline method.

7.1.2 Assessment of Mental Disorder Symptoms

Prior research examining 'mental state detection through speech' has focused on fully supervised learning approaches employing strongly labeled data. However, strong labeling of individuals high in symptoms or state affect in speech audio data is impractical, in part because it is not possible to identify with high confidence which regions of a long speech indicate the person's symptoms of mental disorders. This thesis presents a weakly supervised learning framework for detecting social anxiety and depression from long audio clips. Specifically, it presents a new and simple shallow neural network based feature modeling/engineering technique, named NN2Vec, to generate meaningful feature representation for audio event detection from long weakly labeled audio data by identifying and exploiting the inherent relationship between speakers' vocal states and mental disorder symptoms. Detecting speakers high in social anxiety or depression symptoms using NN2Vec features achieves F-1 scores 17% and 13% higher than those of the best available baselines. In addition, we present a new multiple instance learning adaptation of a BLSTM classifier, named BLSTM-MIL. The presented novel framework of using NN2Vec features with the BLSTM-MIL classifier achieves F-1 scores of 90.1% and 85.44% in detecting speakers high in social anxiety and depression symptoms.

In current clinical practice, assessment of mental disorder symptoms rely on client self-report and clinician judgment, which are vulnerable to social desirability and other subjective biases. Readily accessible, not intrusive or burdensome, and free of extensive equipment, the presented framework is a scalable complement to health-care providers' self-report, interview, and other assessment modalities.

7.1.3 Behavioral Vocal Events Detection

This thesis presents DAVE, the first system which accurately and automatically detects the 5 vocal events of the Cohen-Mansfield inventory. To our knowledge, the automatic detection of verbal events asking for help, sexual verbal advances, cursing with word sense, and repetitive sentence have not been studied. Our solution of questions detection improves the accuracy above the state of art. To solve the detection problems for asking for help, verbal sexual aggression, and questions we use a novel combination of text mining and signal processing.

Cursing is difficult to detect because many such words have multiple meanings. Moreover, there is no existing labeled dataset what contains different meanings of cursing in different contexts. To address this challenge this thesis presents a modified version of the adapted Lesk algorithm [17] which considers a word's sense from a knowledge base, named WordNet, to detect curse words with multiple ambiguous meanings.

DAVE is evaluated on 34 real agitated elderly (age varies from 63 to 98 years) dementia patients across 16 different nursing homes and achieved 90%, 88.1%, 94% and 100% precision for verbal events: asking for help, questions, cursing and asking repetitive sentences, respectively. Moreover we solve the challenge that dementia patients mumble, speak in low volume and don't articulate words well.

7.1.4 Ambient Human Events Detection

This thesis presents a novel framework for ambient human event detection (AHED), which generates robust models for audio monitoring applications with limited available data. Moreover, the generated AHED systems are real-time executable on resource constrained devices. To address the challenge of having limited avilable datasets, we developed an automated audio mixture synthesizer, that can generate a large synthetic dataset with a large variation of background environmental sounds, signal to noise ratios (SNRs), and reverberation effects, from limited available audio samples. Additionally, a computationally effective feature modeling/engineering technique, named Audio2Vec that is robust against environmental noise, reverberation, and de-amplification of sound due to distance is presented. To demonstrate the extensive applicability of the presented AHED framework, we applied the framework to develop and evaluate ambient audio detection models for ten ambient human audio events, and achieved on average 10.3% higher F_1 score compared to the best baseline approaches. To evaluate the applicability of our approach in realistic scenarios, running on resource constrained devices, we implemented an real-time AHED system on a Raspberry Pi 3B with a MATRIX Creator development board. We evaluated the implemented system for two realistic applications: real homes and inside car monitoring. According to the evaluation we achieved average F_1 scores of 0.96 and 0.956 for AHED in real-home and in-car settings, respectively.

7.2 Limitations and Future Improvements

There are some notable extensions and improvements that are possible to the research we have presented in this thesis.

First, all the evaluation presented in this thesis was performed in audio clips or data samples taken from similar distributions. In the future, we look forward to explore the applicability or adaptation of the presented feature modeling/engineering approaches (i.e., Emo2Vec, NN2Vec, Audio2Vec) when audio data source distribution is very different compared to the training audio data source distribution. For example, if we train a model on audio data collected with clean indoor-microphone, how it will perform in extremely noisy YouTube audio clips.

Second, Spontaneous human conversations contain overlapping speech. Currently there is no existing solution to detect emotion or mental disorder from the overlapping portion of speech signal. Our presented solution detects overlapping portion of speech signal, and avoid the portion for verbal

event detection. In future we will investigate the possibility of assessment of mental states (i.e., emotion, disorder) from overlapping portion of speech. Additionally, real environments multiple targeted audio events can occur simultaneously, and their signal may overlap. For example, there can be gunshot when people are screaming. Our solution in chapter 6 did not address such cases.

Third, linguistic content of the spoken utterance is an important part of the conveyed human mental states (i.e., emotion, disorders). But, current speech-to-text transcription approaches still perform poorly when significant noise, reverberation and de-amplification is present in speech. Since, accurate transcription of distance speech signal was out of scope of this study, we focused only on mental state detection from audio signals (in chapter 3 & 4).

In chapter 6 generating some of the ambient human event sounds, such as, gunshot, baby cry were not feasible in real home or car. Hence, we played sounds of targeted events through a Sony SRS-XB10 Bluetooth speaker. In future, we will perform a long term study in the wild of our AHED system, to evaluate it's real-world performance.

The presented study in chapter 4 has several limitations related to sampling. First, we used an analog sample of people high versus low in social anxiety symptoms for whom no formal diagnoses of social anxiety disorder had been established. Second, we analyzed speech audio data from only one situation (a speech stress or task), so future work would benefit from sampling speech from a wider range of both social and nonsocial situations to determine the boundaries of the models' predictive validity.

Moreover, we wish to emphasize that implementation of our approach (in chapter 3, 5 & 4), is designed to support health-care providers and it's practical use must include the informed consent of clients, who should be allowed to discontinue the monitoring at any time, and robust privacy protections. It is important to note that our approach does not use the semantics (transcribed text) of the client's speech and that the proposed feature extraction is irreversible (section 4.1.2), thereby ensuring clients' privacy. Any feedback provided to the client about increases in symptoms would ultimately be paired with treatment resources or other services (e.g., interventions) that the client can use to seek relief.

Additionally, this thesis has not addressed the security and privacy issues of residents (in home-health monitoring settings) in detail. For example, an attacker may compromise the system, or may access the audio data recording device. This is still an open problem and a promising direction for future work.

Finally, we evaluated our presented solutions on real patients' or participants' data (labeled by licensed clinical psychologist or behavioral scientist) but no long-term home health monitoring study was conducted. Future research is needed to evaluate the feasibility, acceptability, and safety of our presented human verbal event monitoring approaches before health-care providers implement the approach on a large scale in the community.

Bibliography

[1] Dragon naturallyspeaking. http://tinyurl.com/26rcknk, Jan 2016.

[2] Tatoeba. https://goo.gl/sr54d0, Jan 2016.

[3] Urban dictionary. http://www.urbandictionary.com/, Jan 2016.

[4] The wiki talk pages. https://en.wikipedia.org/wiki/Help:Using_talk_pages, July 2016.

[5] Google cloud speech api. https://cloud.google.com/speech/, Feb 2017.

[6] Spaces in new homes. goo.gl/1z3oVs, Feb 2017.

[7] spectral noise gating algorithm. http://tinyurl.com/yard8oe, Jan 2017.

[8] Substance Abuse. Mental health services administration.(2018). key substance use and mental health indicators in the united states: Results from the 2017 national survey on drug use and health (hhs publication no. sma 18-5068, nsduh series h-53). rockville, md: Center for behavioral health statistics and quality. *Substance Abuse and Mental Health Services Administration. Retrieved from hhttps://www. samhsa. gov/data/report/2017-nsduh-annual-national-report. Accessed October*, 19, 2018.

[9] Lynn E Alden and Scott T Wallace. Social phobia and social appraisal in successful and unsuccessful social interactions. *Behaviour Research and Therapy*, 33(5):497–505, 1995.

[10] Christos-Nikolaos Anagnostopoulos, Theodoros Iliou, and Ioannis Giannoukos. Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011. *Artificial Intelligence Review*, 43(2):155–177, 2015.

[11] Stuart Andrews, Ioannis Tsochantaridis, and Thomas Hofmann. Support vector machines for multiple-instance learning. In *Advances in neural information processing systems*, pages 577–584, 2003.

[12] Juan Pablo Arias, Carlos Busso, and Nestor Becerra Yoma. Shape-based modeling of the fundamental frequency contour for emotion detection in speech. *Computer Speech & Language*, 28(1):278–294, 2014.

[13] Britt Armour. The Future Of Voice Assistants And AI. `https://clearbridgemobile.com/7-key-predictions-for-the-future-of-voice-assistants-and-ai/`, 2019. [Online; accessed 28-Jan-2019].

[14] American Psychiatric Association et al. *Diagnostic and statistical manual of mental disorders (DSM-5®)*. American Psychiatric Pub, 2013.

[15] Pradeep K Atrey, Namunu C Maddage, and Mohan S Kankanhalli. Audio based event detection for multimedia surveillance. In *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, volume 5, pages V–V. IEEE, 2006.

[16] RG Bachu, S Kopparthi, B Adapa, and BD Barkana. Separation of voiced and unvoiced using zero crossing rate and energy of the speech signal. In *American Society for Engineering Education (ASEE) Zone Conference Proceedings*, pages 1–7, 2008.

[17] Satanjeev Banerjee and Ted Pedersen. An adapted lesk algorithm for word sense disambiguation using wordnet. In *Computational linguistics and intelligent text processing*, pages 136–145. Springer, 2002.

[18] Emilia I Barakova and Tino Lourens. Expressing and interpreting emotional movements in social games with robots. *Personal and ubiquitous computing*, 14(5):457–467, 2010.

[19] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155, 2003.

[20] Kofi Boakye, Benoit Favre, and Dilek Hakkani-Tür. Any questions? automatic question detection in meetings. In *Automatic Speech Recognition & Understanding, 2009. ASRU 2009. IEEE Workshop on*, pages 485–489. IEEE, 2009.

[21] Mehdi Boukhechba, Yu Huang, Philip Chow, Karl Fua, Bethany A Teachman, and Laura E Barnes. Monitoring social anxiety from mobility and communication patterns. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*, pages 749–753. ACM, 2017.

[22] U.S. Census Bureaus. U.S. Census Bureaus 2017. `https://www.census.gov/programs-surveys/popproj.html`, 2019. [Online; accessed 28-Jan-2019].

[23] Michel Cabanac. What is emotion? *Behavioural processes*, 60(2):69–83, 2002.

[24] CPI Inflation Calculator. Bureau of labor statistics, us department of labor, 2016.

[25] Linlin Chao, Jianhua Tao, Minghao Yang, and Ya Li. Improving generation performance of speech emotion recognition by denoising autoencoders. In *Chinese Spoken Language Processing (ISCSLP), 2014 9th International Symposium on*, pages 341–344. IEEE, 2014.

[26] Lijiang Chen, Xia Mao, Yuli Xue, and Lee Lung Cheng. Speech emotion recognition: Features and classification models. *Digital signal processing*, 22(6):1154–1160, 2012.

[27] Zeya Chen, Mohsin Y Ahmed, Asif Salekin, and John A Stankovic. Arasid: Artificial reverberation-adjusted indoor speaker identification dealing with variable distances. In *Proceedings of the 2019 International Conference on Embedded Wireless Systems and Networks*, pages 154–165. Junction Publishing, 2019.

[28] Veronika Cheplygina, David MJ Tax, and Marco Loog. Multiple instance learning with bag dissimilarities. *Pattern Recognition*, 48(1):264–275, 2015.

[29] Philip Chow, Haoyi Xiong, Karl Fua, Wes Bonelli, Bethany A Teachman, and Laura E Barnes. Sad: Social anxiety and depression monitoring system for college students. 2016.

[30] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[31] Chloé Clavel, Thibaut Ehrette, and Gaël Richard. Events detection for an audio-based surveillance system. In *Multimedia and Expo, 2005. ICME 2005. IEEE International Conference on*, pages 1306–1309. IEEE, 2005.

[32] Elise M Clerkin and Bethany A Teachman. Training implicit social anxiety associations: An experimental intervention. *Journal of anxiety disorders*, 24(3):300–308, 2010.

[33] Elise M Clerkin and Bethany A Teachman. Training interpretation biases among individuals with symptoms of obsessive compulsive disorder. *Journal of Behavior Therapy and Experimental Psychiatry*, 42(3):337–343, 2011.

[34] Marti Cleveland-Innes and Prisca Campbell. Emotional presence, learning, and the online learning environment. *The International Review of Research in Open and Distributed Learning*, 13(4):269–292, 2012.

[35] Jiska Cohen-Mansfield. Instruction manual for the cohen-mansfield agitation inventory (cmai). *Research Institute of the Hebrew Home of Greater Washington*, 1991.

[36] Jiska Cohen-Mansfield. Conceptualization of agitation: results based on the cohen-mansfield agitation inventory and the agitation behavior mapping instrument. *International Psychogeriatrics*, 8(S3):309–315, 1997.

[37] Kerry A Collins, Henny A Westra, David JA Dozois, and David D Burns. Gaps in accessing treatment for anxiety and depression: challenges for the delivery of care. *Clinical psychology review*, 24(5):583–616, 2004.

[38] Nicholas Cummins, Julien Epps, Michael Breakspear, and Roland Goecke. An investigation of depressed speech detection: Features and normalization. In *Twelfth Annual Conference of the International Speech Communication Association*, 2011.

[39] Nicholas Cummins, Stefan Scherer, Jarek Krajewski, Sebastian Schnieder, Julien Epps, and Thomas F Quatieri. A review of depression and suicide risk assessment using speech analysis. *Speech Communication*, 71:10–49, 2015.

[40] Nicholas Cummins, Bogdan Vlasenko, Hesam Sagha, and Björn Schuller. Enhancing speech-based depression detection through gender dependent vowel-level formant. In *Proc. of Conference on Artificial Intelligence in Medicine. Springer*, page 5, 2017.

[41] Sally S Dickerson and Margaret E Kemeny. Acute stressors and cortisol responses: a theoretical integration and synthesis of laboratory research. *Psychological bulletin*, 130(3):355, 2004.

[42] Pedro M Domingos. A few useful things to know about machine learning. *Commun. acm*, 55(10):78–87, 2012.

[43] Gary Doran and Soumya Ray. A theoretical and empirical analysis of support vector machine methods for multiple-instance classification. *Machine Learning*, 97(1-2):79–102, 2014.

[44] Moataz El Ayadi, Mohamed S Kamel, and Fakhri Karray. Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognition*, 44(3):572–587, 2011.

[45] Jean Endicott et al. Barriers to seeking treatment for major depression. *Depression and anxiety*, 4(6):273–278, 1996.

[46] C Evers and JR Hopgood. Parametric modelling for single-channel blind dereverberation of speech from a moving speaker. *IET Signal Processing*, 2(2):59–74, 2008.

[47] Christine Evers. Blind dereverberation of speech from moving and stationary speakers using sequential monte carlo methods. 2010.

[48] Florian Eyben, Klaus R Scherer, Björn W Schuller, Johan Sundberg, Elisabeth André, Carlos Busso, Laurence Y Devillers, Julien Epps, Petri Laukka, Shrikanth S Narayanan, et al. The geneva minimalistic acoustic parameter set (gemaps) for voice research and affective computing. *IEEE Transactions on Affective Computing*, 7(2):190–202, 2016.

[49] Florian Eyben, Martin Wöllmer, and Björn Schuller. Opensmile: the munich versatile and fast open-source audio feature extractor. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 1459–1462. ACM, 2010.

[50] Aaron J Fisher. Toward a dynamic model of psychological assessment: Implications for personalized care. *Journal of Consulting and Clinical Psychology*, 83(4):825, 2015.

[51] Alistair J Flint, Sandra E Black, Irene Campbell-Taylor, Gillian F Gailey, and Carey Levinton. Abnormal speech articulation, psychomotor retardation, and subcortical dysfunction in major depression. *Journal of psychiatric research*, 27(3):309–319, 1993.

[52] Pasquale Foggia, Nicolai Petkov, Alessia Saggese, Nicola Strisciuglio, and Mario Vento. Reliable detection of audio events in highly noisy environments. *Pattern Recognition Letters*, 65:22–28, 2015.

[53] Frederic Font, Gerard Roma, and Xavier Serra. Freesound technical demo. In *Proceedings of the 21st acm international conference on multimedia*, pages 411–412. ACM, 2013.

[54] Victor Foo Siang Fook, Pham Viet Thang, That Mon, Qiang Qiu Htwe, Aung Aung Phyo Phyo, Biswas Jit Jayachandran, and Philip Yap. Automated recognition of complex agitation behavior of demented patient using video camera. 2007.

[55] Daniel Joseph France, Richard G Shiavi, Stephen Silverman, Marilyn Silverman, and M Wilkes. Acoustical properties of speech as indicators of depression and suicidal risk. *IEEE transactions on Biomedical Engineering*, 47(7):829–837, 2000.

[56] Daniel Garcia-Romero and Carol Y Espy-Wilson. Analysis of i-vector length normalization in speaker recognition systems. In *Interspeech*, volume 2011, pages 249–252, 2011.

[57] Stephanie Gillespie, Elliot Moore, Jacqueline Laures-Gore, Matthew Farina, Scott Russell, and Yash-Yee Logan. Detecting stress and depression in adults with aphasia through speech analysis. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 5140–5144. IEEE, 2017.

[58] Ofer Golan, Emma Ashwin, Yael Granader, Suzy McClintock, Kate Day, Victoria Leggett, and Simon Baron-Cohen. Enhancing emotion recognition in children with autism spectrum conditions: An intervention using animated vehicles with real emotional faces. *Journal of autism and developmental disorders*, 40(3):269–279, 2010.

[59] Yoav Goldberg and Omer Levy. word2vec explained: Deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

[60] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pages 6645–6649. IEEE, 2013.

[61] Paul Green and Lisa Wei-Haas. The rapid development of user interfaces: Experience with the wizard of oz method. In *Proceedings of the Human Factors Society Annual Meeting*, volume 29, pages 470–474. SAGE Publications Sage CA: Los Angeles, CA, 1985.

[62] Lisa M Haddad and Tammy J Toney-Butler. Nursing shortage. In *StatPearls [Internet]*. StatPearls Publishing, 2019.

[63] Judith A Hall, Jinni A Harrigan, and Robert Rosenthal. Nonverbal behavior in clinician-patient interaction. *Applied and Preventive Psychology*, 4(1):21–37, 1995.

[64] Max Hamilton. A rating scale for depression. *Journal of neurology, neurosurgery, and psychiatry*, 23(1):56, 1960.

[65] Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

[66] S. Haq and P.J.B. Jackson. *Machine Audition: Principles, Algorithms and Systems*, chapter Multimodal Emotion Recognition, pages 398–423. IGI Global, Hershey PA, Aug. 2010.

[67] Robert Hecht-Nielsen et al. Theory of the backpropagation neural network. *Neural Networks*, 1(Supplement-1):445–448, 1988.

[68] Richard G Heimberg, Craig S Holt, Franklin R Schneier, Robert L Spitzer, and Michael R Liebowitz. The issue of subtypes in the diagnosis of social phobia. *Journal of Anxiety Disorders*, 7(3):249–269, 1993.

[69] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[70] Richard Hodach, Alide Chase, Robert Fortini, Connie Delaney, and Richard Hodach. Population health management: a roadmap for provider-based automation in a new era of healthcare. *Institute for Health Technology Transformation. Retrieved from*, 2012.

[71] James R Hopgood and Christine Evers. Block-based tvar models for single-channel blind dereverberation of speech from a moving speaker. In *Statistical Signal Processing, 2007. SSP'07. IEEE/SP 14th Workshop on*, pages 274–278. IEEE, 2007.

[72] Yu Huang, Jiaqi Gong, Mark Rucker, Philip Chow, Karl Fua, Matthew S Gerber, Bethany Teachman, and Laura E Barnes. Discovery of behavioral markers of social anxiety from smartphone sensor data. In *Proceedings of the 1st Workshop on Digital Biomarkers*, pages 9–14. ACM, 2017.

[73] Zhengwei Huang, Ming Dong, Qirong Mao, and Yongzhao Zhan. Speech emotion recognition using cnn. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 801–804. ACM, 2014.

[74] Hussein Hussein, Marc Ritter, Robert Manthey, Jan Schloßhauer, Etienne Fabian, and Manuel Heinzig. Acoustic event classification for ambient assisted living and healthcare environments. In *Proceedings of the 27th Conference on Electronic Speech Signal Processing (ESSV)*, volume 81, pages 271–278.

[75] Mark Jager, Christian Knoll, and Fred A Hamprecht. Weakly supervised learning of a classifier for unusual event detection. *IEEE Transactions on Image Processing*, 17(9):1700–1708, 2008.

[76] Christian Jones and Jamie Sutherland. Acoustic emotion recognition for affective computer gaming. In *Affect and emotion in human-computer interaction*, pages 209–219. Springer, 2008.

[77] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*, 2016.

[78] Daniel Jurafsky, Rebecca Bates, Noah Coccaro, Rachel Martin, Marie Meteer, Klaus Ries, Elizabeth Shriberg, Andreas Stolcke, Paul Taylor, and Carol Van Ess-Dykema. Automatic detection of discourse structure for speech recognition and understanding. In *Automatic Speech Recognition and Understanding, 1997. Proceedings., 1997 IEEE Workshop on*, pages 88–95. IEEE, 1997.

[79] Martin Karafiát, Lukáš Burget, Pavel Matějka, Ondřej Glembek, and Jan Černockỳ. ivector-based discriminative adaptation for automatic speech recognition. In *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*, pages 152–157. IEEE, 2011.

[80] Lakshmish Kaushik, Abhijeet Sangwan, and John HL Hansen. Laughter and filler detection in naturalistic audio. In *INTERSPEECH*, pages 2509–2513, 2015.

[81] Alan E Kazdin. Addressing the treatment gap: A key challenge for extending evidence-based psychosocial interventions. *Behaviour research and therapy*, 88:7–18, 2017.

[82] Patrick Kenny, Gilles Boulianne, and Pierre Dumouchel. Eigenvoice modeling with sparse training data. *IEEE transactions on speech and audio processing*, 13(3):345–354, 2005.

[83] Kasturi Rangan Krishnamachari, Robert E Yantorno, Jereme M Lovekin, Daniel S Benincasa, and Stanley J Wenndt. Use of local kurtosis measure for spotting usable speech segments in co-channel speech. In *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, volume 1, pages 649–652. IEEE, 2001.

[84] Kurt Kroenke, Tara W Strine, Robert L Spitzer, Janet BW Williams, Joyce T Berry, and Ali H Mokdad. The phq-8 as a measure of current depression in the general population. *Journal of affective disorders*, 114(1):163–173, 2009.

[85] Anurag Kumar, Rajesh M Hegde, Rita Singh, and Bhiksha Raj. Event detection in short duration audio using gaussian mixture model and random forest classifier. In *Signal Processing Conference (EUSIPCO), 2013 Proceedings of the 21st European*, pages 1–5. IEEE, 2013.

[86] Anurag Kumar and Bhiksha Raj. Audio event detection using weakly labeled data. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 1038–1047. ACM, 2016.

[87] Anurag Kumar and Bhiksha Raj. Weakly supervised scalable audio content analysis. In *Multimedia and Expo (ICME), 2016 IEEE International Conference on*, pages 1–6. IEEE, 2016.

[88] Anurag Kumar and Bhiksha Raj. Deep cnn framework for audio event recognition using weakly labeled web data. *arXiv preprint arXiv:1707.02530*, 2017.

[89] Petri Laukka, Clas Linnman, Fredrik Åhs, Anna Pissiota, Örjan Frans, Vanda Faria, Åsa Michelgård, Lieuwe Appel, Mats Fredrikson, and Tomas Furmark. In a nervous voice: Acoustic analysis and perception of anxiety in social phobics' speech. *Journal of Nonverbal Behavior*, 32(4):195, 2008.

[90] Yizhar Lavner, Rami Cohen, Dima Ruinskiy, and Hans IJzerman. Baby cry detection in domestic environment using deep learning. In *Science of Electrical Engineering (ICSEE), IEEE International Conference on the*, pages 1–5. IEEE, 2016.

[91] Duc Le, Zakaria Aldeneh, and Emily Mower Provost. Discretized continuous speech emotion recognition with multi-task deep recurrent neural network. *Interspeech, 2017 (to apear)*, 2017.

[92] Duc Le and Emily Mower Provost. Emotion recognition from spontaneous speech using hidden markov models with deep belief networks. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 216–221. IEEE, 2013.

[93] Sungbok Lee, Serdar Yildirim, Abe Kazemzadeh, and Shrikanth Narayanan. An articulatory study of emotional speech production. In *Interspeech*, pages 497–500, 2005.

[94] ZHOU Lian. Exploration of the working principle and application of word2vec. *Sci-Tech Information Development & Economy*, 2:145–148, 2015.

[95] Wootaek Lim, Daeyoung Jang, and Taejin Lee. Speech emotion recognition using convolutional and recurrent neural networks. In *Signal and information processing association annual summit and conference (APSIPA), 2016 Asia-Pacific*, pages 1–4. IEEE, 2016.

[96] Paula Lopez-Otero, Laura Docio-Fernandez, and Carmen Garcia-Mateo. ivectors for continuous emotion recognition. *Training*, 45:50, 2014.

[97] Wladyslaw Losiak, Agata Blaut, Joanna Klosowska, and Natalia Slowik. Social anxiety, affect, cortisol response and performance on a speech task. *Psychopathology*, 49(1):24–30, 2016.

[98] Lu-Shih Alex Low, Namunu C Maddage, Margaret Lech, Lisa B Sheeber, and Nicholas B Allen. Detection of clinical depression in adolescents' speech during family interactions. *IEEE Transactions on Biomedical Engineering*, 58(3):574–586, 2011.

[99] Harold Lunenfeld. Human factor considerations of motorist navigation and information systems. In *Vehicle Navigation and Information Systems Conference, 1989. Conference Record*, pages 35–42. IEEE, 1989.

[100] Xingchen Ma, Hongyu Yang, Qiang Chen, Di Huang, and Yunhong Wang. Depaudionet: An efficient deep model for audio based depression classification. In *Proceedings of the 6th International Workshop on Audio/Visual Emotion Challenge*, pages 35–42. ACM, 2016.

[101] Lenka Macková, Anton Čižmár, and Jozef Juhár. Emotion recognition in i-vector space. In *Radioelektronika (RADIOELEKTRONIKA), 2016 26th International Conference*, pages 372–375. IEEE, 2016.

[102] Erik Marchi, Giacomo Ferroni, Florian Eyben, Leonardo Gabrielli, Stefano Squartini, and Bjorn Schuller. Multi-resolution linear prediction based features for audio onset detection with bidirectional lstm neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 2164–2168. IEEE, 2014.

[103] Anna Margolis and Mari Ostendorf. Question detection in spoken conversations using textual conversations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 118–124. Association for Computational Linguistics, 2011.

[104] MATRIX. MATRIX Creator. `https://www.matrix.one/products/creator`, 2019. [Online; accessed 28-Jan-2019].

[105] Richard P Mattick and J Christopher Clarke. Development and validation of measures of social phobia scrutiny fear and social interaction anxiety. *Behaviour research and therapy*, 36(4):455–470, 1998.

[106] Mary L McHugh. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3):276–282, 2012.

[107] Ian McLoughlin, Haomin Zhang, Zhipeng Xie, Yan Song, and Wei Xiao. Robust sound event classification using deep neural networks. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):540–552, 2015.

[108] Philip Mendes, Badal Moslehuddin, Chris Goddard, et al. Improving the physical and emotional health of young people transitioning from state out-of-home care. *Developing practice: the child, youth and family work journal*, (20):33, 2008.

[109] mental health news, 2019. https://www.mentalhealthfirstaid.org/2019/02/5-surprising-mental-health-statistics/.

[110] Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. Tut database for acoustic scene classification and sound event detection. In *Signal Processing Conference (EUSIPCO), 2016 24th European*, pages 1128–1132. IEEE, 2016.

[111] Tomáš Mikolov. Statistical language models based on neural networks. *Presentation at Google, Mountain View, 2nd April*, 2012.

[112] Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Interspeech*, volume 2, page 3, 2010.

[113] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[114] George A Miller. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[115] Alex J Mitchell, Amol Vaze, and Sanjay Rao. Clinical diagnosis of depression in primary care: a meta-analysis. *The Lancet*, 374(9690):609–619, 2009.

[116] Torin Monahan and Tyler Wall. Somatic surveillance: Corporeal control through information networks. *Surveillance & Society*, 4(3), 2002.

[117] Veronica Morfi and Dan Stowell. Data-efficient weakly supervised learning for low-resource audio event detection using deep learning. *arXiv preprint arXiv:1807.06972*, 2018.

[118] James C Mundt, Peter J Snyder, Michael S Cannizzaro, Kara Chappie, and Dayna S Geralts. Voice acoustic measures of depression severity and treatment response collected via interactive voice response (ivr) technology. *Journal of neurolinguistics*, 20(1):50–64, 2007.

[119] Satoshi Nakamura, Kazuo Hiyane, Futoshi Asano, Takanobu Nishiura, and Takeshi Yamada. Acoustical sound database in real environments for sound scene understanding and hands-free speech recognition. In *LREC*, 2000.

[120] Satoshi Nakamura, Kazuo Hiyane, Futoshi Asano, Takeshi Yamada, and Takashi Endo. Data collection in real acoustical environments for sound scene understanding and hands-free speech recognition. 1999.

[121] Mahesh Kumar Nandwana and Taufiq Hasan. Towards smart-cars that can listen: Abnormal acoustic event detection on the road. In *INTERSPEECH*, pages 2968–2971, 2016.

[122] Md Nasir, Arindam Jati, Prashanth Gurunath Shivakumar, Sandeep Nallan Chakravarthula, and Panayiotis Georgiou. Multimodal and multiresolution depression detection from speech and facial landmark features. In *Proceedings of the 6th International Workshop on Audio/Visual Emotion Challenge*, pages 43–50. ACM, 2016.

[123] Shahriar Nirjon, Ifat Afrin Emi, Md Abu Sayeed Mondol, Asif Salekin, and John A Stankovic. Mobi-cog: a mobile application for instant screening of dementia using the mini-cog test. In *Proceedings of the Wireless Health 2014 on National Institutes of Health*, pages 1–7. ACM, 2014.

[124] Shahriar Nirjon, Chris Greenwood, Carlos Torres, Stefanie Zhou, John A Stankovic, Hee Jung Yoon, Ho-Kyeong Ra, Can Basaran, Taejoon Park, and Sang H Son. Kintense: A robust, accurate, real-time and evolving system for detecting aggressive actions from streaming 3d skeleton data. In *Pervasive Computing and Communications (PerCom), 2014 IEEE International Conference on*, pages 2–10. IEEE, 2014.

[125] Stavros Ntalampiras, Ilyas Potamitis, and Nikos Fakotakis. An adaptive framework for acoustic monitoring of potential hazards. *EURASIP Journal on Audio, Speech, and Music Processing*, 2009(1):594103, 2009.

[126] Office of Special Education and Rehabilitative Services (ED). *37th Annual Report to Congress on the Implementation of the" Individuals with Disabilities Education Act," 2015*. ERIC Clearinghouse, 2015.

[127] Mark Olfson, Mary Guardino, Elmer Struening, Franklin R Schneier, Fred Hellman, and Donald F Klein. Barriers to the treatment of social anxiety. *American Journal of Psychiatry*, 157(4):521–527, 2000.

[128] Luiza Orosanu and Denis Jouvet. Combining lexical and prosodic features for automatic detection of sentence modality in french. In *International Conference on Statistical Language and Speech Processing*, pages 207–218. Springer, 2015.

[129] Stephanie Pancoast and Murat Akbacak. Bag-of-audio-words approach for multimedia event classification. In *Interspeech*, pages 2105–2108, 2012.

[130] Giambattista Parascandolo, Toni Heittola, Heikki Huttunen, Tuomas Virtanen, et al. Convolutional recurrent neural networks for polyphonic sound event detection. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(6):1291–1303, 2017.

[131] Giambattista Parascandolo, Heikki Huttunen, and Tuomas Virtanen. Recurrent neural networks for polyphonic sound event detection in real life recordings. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 6440–6444. IEEE, 2016.

[132] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. *ICML (3)*, 28:1310–1318, 2013.

[133] Vikram Patel and RA Hope. A rating scale for aggressive behaviour in the elderly–the rage. *Psychological medicine*, 22(01):211–221, 1992.

[134] Jian Pei, Jiawei Han, Behzad Mortazavi-Asl, Jianyong Wang, Helen Pinto, Qiming Chen, Umeshwar Dayal, and Mei-Chun Hsu. Mining sequential patterns by pattern-growth: The prefixspan approach. *Knowledge and Data Engineering, IEEE Transactions on*, 16(11):1424–1440, 2004.

[135] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[136] Rosalind W Picard. Toward computers that recognize and respond to user emotion. *IBM systems journal*, 39(3.4):705–719, 2000.

[137] Karol J Piczak. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1015–1018. ACM, 2015.

[138] Karol J Piczak. Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1015–1018. ACM, 2015.

[139] Oudeyer Pierre-Yves. The production and recognition of emotions in speech: features and algorithms. *International Journal of Human-Computer Studies*, 59(1):157–183, 2003.

[140] Tony Plate. Distributed representations. *Encyclopedia of Cognitive Science*.

[141] Tim Polzehl, Alexander Schmitt, Florian Metze, and Michael Wagner. Anger recognition in speech using acoustic and linguistic cues. *Speech Communication*, 53(9):1198–1209, 2011.

[142] V Minh Quang, Laurent Besacier, and Eric Castelli. Automatic question detection: prosodic-lexical features and cross-lingual experiments. In *Proc. Interspeech*, volume 2007, pages 2257–2260, 2007.

[143] Vũ Minh Quang, Eric Castelli, and Phm Ngc Yên. A decision tree-based method for speech processing: question sentence detection. In *International Conference on Fuzzy Systems and Knowledge Discovery*, pages 1205–1212. Springer, 2006.

[144] Ho-Kyeong Ra, Asif Salekin, Hee Jung Yoon, Jeremy Kim, Shahriar Nirjon, David J Stone, Sujeong Kim, Jong-Myung Lee, Sang Hyuk Son, and John A Stankovic. Asthmaguide: an asthma monitoring and advice ecosystem. In *2016 IEEE Wireless Health (WH)*, pages 1–8. IEEE, 2016.

[145] Ho-Kyeong Ra, Hee Jung Yoon, Asif Salekin, Jin-Hee Lee, John A Stankovic, and Sang Hyuk Son. Poster: Software architecture for efficiently designing cloud applications using node. js. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services Companion*, pages 72–72. ACM, 2016.

[146] S Ramakrishnan and Ibrahiem MM El Emary. Speech emotion recognition approaches in human computer interaction. *Telecommunication Systems*, pages 1–12, 2013.

[147] Rajib Rana. Emotion classification from noisy speech-a deep learning approach. *arXiv preprint arXiv:1603.05901*, 2016.

[148] K Sreenivasa Rao, Tummala Pavan Kumar, Kusam Anusha, Bathina Leela, Ingilela Bhavana, and SVSK Gowtham. Emotion recognition from speech. *International Journal of Computer Science and Information Technologies*, 3(2):3603–3607, 2012.

[149] Ronald M Rapee and Lina Lim. Discrepancy between self-and observer ratings of performance in social phobics. *Journal of abnormal psychology*, 101(4):728, 1992.

[150] Shourabh Rawat, Peter F Schulam, Susanne Burger, Duo Ding, Yipei Wang, and Florian Metze. Robust audio-codebooks for large-scale event detection in consumer videos. 2013.

[151] Steve Renals and Pawel Swietojanski. Neural networks for distant speech recognition. In *Hands-free Speech Communication and Microphone Arrays (HSCMA), 2014 4th Joint Workshop on*, pages 172–176. IEEE, 2014.

[152] Steven A Rieger, Rajani Muraleedharan, and Ravi P Ramachandran. Speech based emotion recognition using spectral feature extraction and an ensemble of knn classifiers. In *Chinese Spoken Language Processing (ISCSLP), 2014 9th International Symposium on*, pages 589–593. IEEE, 2014.

[153] Xin Rong. word2vec parameter learning explained. *arXiv preprint arXiv:1411.2738*, 2014.

[154] J-L Rouas, Jérôme Louradour, and Sébastien Ambellouis. Audio events detection in public transport vehicle. In *Intelligent Transportation Systems Conference, 2006. ITSC'06. IEEE*, pages 733–738. IEEE, 2006.

[155] Melissa Ryan, Janice Murray, and Ted Ruffman. Aging and the perception of emotion: Processing vocal expressions alone and with faces. *Experimental aging research*, 36(1):1–22, 2009.

[156] Tara N Sainath, Brian Kingsbury, George Saon, Hagen Soltau, Abdel-rahman Mohamed, George Dahl, and Bhuvana Ramabhadran. Deep convolutional neural networks for large-scale speech tasks. *Neural Networks*, 64:39–48, 2015.

[157] Justin Salamon, Christopher Jacoby, and Juan Pablo Bello. A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 1041–1044. ACM, 2014.

[158] Asif Salekin, Zeya Chen, Mohsin Y Ahmed, John Lach, Donna Metz, Kayla De La Haye, Brooke Bell, and John A Stankovic. Distant emotion recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 1(3):96, 2017.

[159] Asif Salekin, Jeremy W Eberle, Jeffrey J Glenn, Bethany A Teachman, and John A Stankovic. A weakly supervised learning framework for detecting social anxiety and depression. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2(2):81, 2018.

[160] Asif Salekin, Shabnam Ghaffarzadegan, Zhe Feng, and John Stankovic. A real-time audio monitoring framework with limited data for constrained devices. In *Proceedings of the 15th International Conference on Distributed Computing in Sensor Systems*. IEEE, 2019.

[161] Asif Salekin and John Stankovic. Detection of chronic kidney disease and selecting important predictive attributes. In *2016 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 262–270. IEEE, 2016.

[162] Asif Salekin, Hongning Wang, and John Stankovic. Kinvocal: Detecting agitated vocal events. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, pages 459–460. ACM, 2015.

[163] Asif Salekin, Hongning Wang, Kristine Williams, and John Stankovic. Dave: detecting agitated vocal events. In *Proceedings of the Second IEEE/ACM International Conference on Connected Health: Applications, Systems and Engineering Technologies*. IEEE Press, 2017.

[164] Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

[165] Ankur Sapra, Nikhil Panwar, and Sohan Panwar. Emotion recognition from speech. *International Journal of Emerging Technology and Advanced Engineering*, 3:341–345, 2013.

[166] Stefan Scherer, Louis-Philippe Morency, Jonathan Gratch, and John Pestian. Reduced vowel space is a robust indicator of psychological distress: a cross-corpus analysis. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 4789–4793. IEEE, 2015.

[167] Stefan Scherer, Giota Stratou, Jonathan Gratch, and Louis-Philippe Morency. Investigating voice quality as a speaker-independent indicator of depression and ptsd. In *Interspeech*, pages 847–851, 2013.

[168] Annett Schirmer and Ralph Adolphs. Emotion perception from face, voice, and touch: comparisons and convergence. *Trends in cognitive sciences*, 21(3):216–228, 2017.

[169] M Schroder and R Cowie. Issues in emotion-oriented computing toward a shared understanding. In *Workshop on emotion and computing*, 2006.

[170] Björn Schuller, Stefan Steidl, Anton Batliner, Alessandro Vinciarelli, Klaus Scherer, Fabien Ringeval, Mohamed Chetouani, Felix Weninger, Florian Eyben, Erik Marchi, et al. The interspeech 2013 computational paralinguistics challenge: social signals, conflict, emotion, autism. 2013.

[171] Margaret C Sewell, Xiaodong Luo, Judith Neugroschl, and Mary Sano. Detection of mild cognitive impairment and early stage dementia with an audio-recorded cognitive scale. *International Psychogeriatrics*, 25(08):1325–1333, 2013.

[172] Roneel V Sharan and Tom J Moir. Comparison of multiclass svm classification techniques in an audio surveillance application under mismatched conditions. In *Digital Signal Processing (DSP), 2014 19th International Conference on*, pages 83–88. IEEE, 2014.

[173] Navid Shokouhi, Amardeep Sathyanarayana, Seyed Omid Sadjadi, and John HL Hansen. Overlapped-speech detection with applications to driver assessment for in-vehicle active safety systems. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 2834–2838. IEEE, 2013.

[174] Diana Sidtis and Jody Kreiman. In the beginning was the familiar voice: personally familiar voices in the evolutionary and contemporary biology of communication. *Integrative Psychological and Behavioral Science*, 46(2):146–159, 2012.

[175] Vered Silber-Varod, Hamutal Kreiner, Ronen Lovett, Yossi Levi-Belz, and Noam Amir. Do social anxiety individuals hesitate more? the prosodic profile of hesitation disfluencies in social

anxiety disorder individuals. *Proceedings of Speech Prosody 2016 (SP2016)*, pages 1211–1215, 2016.

[176] Christina Sobin and Harold A Sackeim. Psychomotor symptoms of depression. *The American journal of psychiatry*, 154(1):4, 1997.

[177] Donald F Specht. Probabilistic neural networks. *Neural networks*, 3(1):109–118, 1990.

[178] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[179] Brian Stasak, Julien Epps, Nicholas Cummins, and Roland Goecke. An investigation of emotional speech in depression classification. In *INTERSPEECH*, pages 485–489, 2016.

[180] Greg J Stephens, Lauren J Silbert, and Uri Hasson. Speaker–listener neural coupling underlies successful communication. *Proceedings of the National Academy of Sciences*, 107(32):14425–14430, 2010.

[181] Melissa N Stolar, Margaret Lech, and Nicholas B Allen. Detection of depression in adolescents based on statistical modeling of emotional influences in parent-adolescent conversations. In *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*, pages 987–991. IEEE, 2015.

[182] Lusia Stopa and David M Clark. Cognitive processes in social phobia. *Behaviour Research and Therapy*, 31(3):255–267, 1993.

[183] Ting-Wei Su, Jen-Yu Liu, and Yi-Hsuan Yang. Weakly-supervised audio event detection using event-specific gaussian filters and fully convolutional networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 791–795. IEEE, 2017.

[184] James E Swain, Esra Tasgin, Linda C Mayes, Ruth Feldman, R Todd Constable, and James F Leckman. Maternal brain response to own baby-cry is affected by cesarean section delivery. *Journal of child psychology and psychiatry*, 49(10):1042–1052, 2008.

[185] Naoya Takahashi, Michael Gygli, Beat Pfister, and Luc Van Gool. Deep convolutional neural networks and data augmentation for acoustic event detection. *arXiv preprint arXiv:1604.07160*, 2016.

[186] Yaodong Tang, Yuchen Huang, Zhiyong Wu, Helen Meng, Mingxing Xu, and Lianhong Cai. Question detection from acoustic features using recurrent neural network with gated recurrent unit. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6125–6129. IEEE, 2016.

[187] Ashish Tawari and Mohan M Trivedi. Speech emotion analysis in noisy real-world environment. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 4605–4608. IEEE, 2010.

[188] Bethany A Teachman. No appointment necessary: Treating mental illness outside the therapist's office. *Perspectives on Psychological Science*, 9(1):85–87, 2014.

[189] Sara Thomée, Annika Härenstam, and Mats Hagberg. Mobile phone use and stress, sleep disturbances, and symptoms of depression among young adults-a prospective cohort study. *BMC public health*, 11(1):66, 2011.

[190] Peter Transfeld, Simon Receveur, and Tim Fingscheidt. An acoustic event detection framework and evaluation metric for surveillance in cars. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[191] George Trigeorgis, Fabien Ringeval, Raymond Brueckner, Erik Marchi, Mihalis A Nicolaou, Björn Schuller, and Stefanos Zafeiriou. Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5200–5204. IEEE, 2016.

[192] Michel Vacher, Dan Istrate, Laurent Besacier, Jean-François Serignat, and Eric Castelli. Sound detection and classification for medical telesurvey. In *2nd Conference on Biomedical Engineering*, pages 395–398, 2004.

[193] Michel Valstar, Jonathan Gratch, Björn Schuller, Fabien Ringeval, Dennis Lalanne, Mercedes Torres Torres, Stefan Scherer, Giota Stratou, Roddy Cowie, and Maja Pantic. Avec 2016: Depression, mood, and emotion recognition workshop and challenge. In *Proceedings of the 6th International Workshop on Audio/Visual Emotion Challenge*, pages 3–10. ACM, 2016.

[194] Bogdan Vlasenko, Hesam Sagha, Nicholas Cummins, and Björn Schuller. Implementing gender-dependent vowel-level analysis for boosting speech-based depression recognition. *Proc. Interspeech 2017*, pages 3266–3270, 2017.

[195] Sabrina C Voci, Joseph H Beitchman, EB Brownlie, and Beth Wilson. Social anxiety in late adolescence: The importance of early childhood language impairment. *Journal of anxiety disorders*, 20(7):915–930, 2006.

[196] Philip S Wang, Patricia Berglund, Mark Olfson, Harold A Pincus, Kenneth B Wells, and Ronald C Kessler. Failure and delay in initial treatment contact after first onset of mental disorders in the national comorbidity survey replication. *Archives of general psychiatry*, 62(6):603–613, 2005.

[197] Wenbo Wang, Lu Chen, Krishnaprasad Thirunarayan, and Amit P Sheth. Cursing in english on twitter. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 415–425. ACM, 2014.

[198] Xinggang Wang, Yongluan Yan, Peng Tang, Xiang Bai, and Wenyu Liu. Revisiting multiple instance neural networks. *arXiv preprint arXiv:1610.02501*, 2016.

[199] Yun Wang, Leonardo Neves, and Florian Metze. Audio-based multimedia event detection using deep recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 2742–2746. IEEE, 2016.

[200] Justin W Weeks, Chao-Yang Lee, Alison R Reilly, Ashley N Howell, Christopher France, Jennifer M Kowalsky, and Ashley Bush. "the sound of fear": Assessing vocal fundamental frequency as a physiological indicator of social anxiety disorder. *Journal of anxiety disorders*, 26(8):811–822, 2012.

[201] Justin W Weeks, Akanksha Srivastav, Ashley N Howell, and Andrew R Menatti. "speaking more than words": Classifying men with social anxiety disorder via vocal acoustic analyses of diagnostic interviews. *Journal of Psychopathology and Behavioral Assessment*, 38(1):30–41, 2016.

[202] E Weiller, J-C Bisserbe, P Boyer, J-P Lepine, and Y Lecrubier. Social phobia in general health care: an unrecognised undertreated disabling disorder. *The British Journal of Psychiatry*, 168(2):169–174, 1996.

[203] Adrian Wells, David M Clark, Paul Salkovskis, John Ludgate, Ann Hackmann, and Michael Gelder. Social phobia: The role of in-situation safety behaviors in maintaining anxiety and negative beliefs. *Behavior Therapy*, 26(1):153–161, 1995.

[204] Pauline K Wiener, Dimitris N Kiosses, Sibel Klimstra, Christopher Murphy, and George S Alexopoulos. A short-term inpatient program for agitated demented nursing home residents. *International journal of geriatric psychiatry*, 16(9):866–872, 2001.

[205] Herman R Bossen A Williams K, Perkhounkova Y. A communication intervention to reduce resistiveness in dementia care: A cluster randomized controlled trial. In *The Gerontologist*.

[206] Chung-Hsien Wu and Wei-Bin Liang. Emotion recognition of affective speech based on multiple classifiers using acoustic-prosodic information and semantic labels. *IEEE Transactions on Affective Computing*, 2(1):10–21, 2011.

[207] Jiajun Wu, Yinan Yu, Chang Huang, and Kai Yu. Deep multiple instance learning for image classification and auto-annotation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3460–3469, 2015.

[208] Rui Xia and Yang Liu. Using i-vector space model for emotion recognition. In *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[209] Bai Xue, Chen Fu, and Zhan Shaobin. A study on sentiment computing and classification of sina weibo with word2vec. In *Big Data (BigData Congress), 2014 IEEE International Congress on*, pages 358–363. IEEE, 2014.

[210] Show-Jane Yen and Yue-Shi Lee. Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, 36(3):5718–5727, 2009.

[211] Takuya Yoshioka, Xie Chen, and Mark JF Gales. Impact of single-microphone dereverberation on dnn-based meeting transcription systems. In *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pages 5527–5531. IEEE, 2014.

[212] Mingyu You, Chun Chen, Jiajun Bu, Jia Liu, and Jianhua Tao. Emotion recognition from noisy speech. In *Multimedia and Expo, 2006 IEEE International Conference on*, pages 1653–1656. IEEE, 2006.

[213] Yael Zemack-Rugar, James R Bettman, and Gavan J Fitzsimons. The effects of nonconsciously priming emotion concepts on behavior. *Journal of Personality and Social Psychology*, 93(6):927, 2007.

[214] Zhihong Zeng, Maja Pantic, Glenn I Roisman, and Thomas S Huang. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE transactions on pattern analysis and machine intelligence*, 31(1):39–58, 2009.

[215] Teng Zhang and Ji Wu. Speech emotion recognition with i-vector feature and rnn model. In *Signal and Information Processing (ChinaSIP), 2015 IEEE China Summit and International Conference on*, pages 524–528. IEEE, 2015.

[216] Wan Li Zhang, Guo Xin Li, and Wei Gao. The research of speech emotion recognition based on gaussian mixture model. In *Applied Mechanics and Materials*, volume 668, pages 1126–1129. Trans Tech Publ, 2014.

[217] Yu Zhang, Guoguo Chen, Dong Yu, Kaisheng Yaco, Sanjeev Khudanpur, and James Glass. Highway long short-term memory rnns for distant speech recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*, pages 5755–5759. IEEE, 2016.

[218] Patrick H Zimmerman, J Elizabeth Bolhuis, Albert Willemsen, Erik S Meyer, and Lucas PJJ Noldus. The observer xt: A tool for the integration and synchronization of multimodal signals. *Behavior research methods*, 41(3):731–735, 2009.