Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Conversational Language Learning for Human-Robot Interaction

Dissertation
submitted to the
Faculty of Mathematics, Informatics and
Natural Sciences (MIN-Faculty),
Department of Informatics,
Universität Hamburg

in partial fulfilment of the
requirements for the degree of
Doctor rerum naturalium (Dr. rer. nat.)

Chandrakant Ramesh Bothe
Hamburg, 2020

Publication Identifier:
urn:nbn:de:gbv:18-ediss-90853

Submission of the thesis:
18th of August 2020

Date of oral defense:
18th of November 2020

Dissertation Committee:
Prof. Dr. Chris Biemann (reviewer)
Department of Computer Science,
Universität Hamburg, Germany

Prof. Dr. Wolfgang Menzel (chair)
Department of Computer Science,
Universität Hamburg, Germany

Prof. Dr. Stefan Wermter (reviewer, advisor)
Department of Computer Science,
Universität Hamburg, Germany

Abstract

Language is one of the complex but fascinating ways of communication, and it is continuously developed and maintained in the human brain. It is remarkable to study how humans understand each other in a conversation and continually learn and develop their communication skills. Understanding the meaning of the spoken or written language and interacting in that language differentiates humans from other species. Although it is difficult to define the exact working nature of the brain related to language acquisition and development, researchers find a strong relationship between different behaviours acquired based on social, cognitive, emotional and behavioural intelligence. Social robots and artificial human-like intelligent agents are the expected members of future society, where they are firmly expected to realize and exhibit verbal communication capability. In addition to the robot appearance, conversational understanding and behaviours are crucial aspects for their acceptance and co-existence in emerging society.

This thesis aims to connect the knowledge from behavioural intelligence through conversational language learning with human-robot interaction (HRI). The socio-linguistic features, such as emotion, sentiment, politeness and dialogue acts, are the building blocks of the decision-making process in humans. This thesis presents extensive conversational analysis through artificial recurrent neural modelling that helps to build the robots aware of such linguistic cues. Accordingly, the thesis provides tools to analyze and investigate language on different aspects using recurrent neural networks (RNNs) and attention mechanism and eventually demonstrates an HRI scenario that facilitates robotics behavioural adaptation based on social cues. As a result, the thesis provides insights into the conversational analysis with emotion and dialogue acts, providing useful knowledge of natural language understanding for safe human-robot interaction.

The primary contribution to knowledge from the study and experiments provided in this thesis is understanding the socio-linguistic features, with the motive of developing a natural language conversational system for HRI. The analytical experiments in this thesis can inform necessary future work in order to integrate social cues for robotic behavioural adaptation. Furthermore, this thesis provides knowledge to realize safer social robots in society with verbal communication capability using computational neural linguistics approaches, along with addressing the safety concerns of humans.

Zusammenfassung

Sprache ist eine komplexe und faszinierende Art der Kommunikation, die sich im menschlichen Gehirn ständig weiterentwickelt und verändert. Es ist bemerkenswert, wie Menschen sich in einem Gespräch verstehen und ihre Kommunikationsfähigkeiten erlernen und kontinuierlich weiterentwickeln. Die Bedeutung gesprochener oder geschriebener Sprache zu verstehen und damit zu interagieren, unterscheidet den Menschen von anderen Spezies. Obwohl es schwierig ist, die genaue Funktionsweise des Gehirns im Zusammenhang mit Spracherwerb und Sprachentwicklung zu definieren, finden Forscher starke Beziehungen zwischen verschiedenen Verhaltensweisen, die auf sozialer, kognitiver, emotionaler und Verhaltensintelligenz beruhen. Von sozialen Robotern und anderen künstlichen menschenähnlichen Agenten wird erwartet, dass sie verbale Kommunikationsfähigkeiten durch Interaktion automatisch erlernen. Neben dem Erscheinungsbild der Roboter sind Gesprächsverständnis und Verhaltensweisen entscheidende Aspekte für ihre Akzeptanz in der Gesellschaft.

Diese Arbeit zielt darauf ab, Forschungsergebnisse aus der Verhaltensintelligenz bezüglich des Erlernens der Konversationssprache mit der Erforschung der Mensch-Roboter-Interaktion (HRI) zu verbinden. Soziolinguistische Merkmale wie Emotion, Gefühl, Höflichkeit und Dialogakte sind die Bausteine des Entscheidungsprozesses des Menschen. Damit Roboter lernen können, solche Merkmale zu nutzen, wird in dieser Arbeit eine umfassende Gesprächsanalyse durch künstliche rekurrente neuronale Netzwerke vorgestellt. Dementsprechend präsentiert diese Arbeit Werkzeuge zur Analyse und Untersuchung von Sprache auf verschiedene Aspekte auf Basis rekurrenter neuronaler Netzwerke (RNNs) und einem Attention-Mechanismus und zeigt letztendlich ein HRI-Szenario, welches die Verhaltensanpassung des Roboters auf Grundlage sozialer Merkmale ermöglicht. Als Ergebnis bietet die Arbeit einen tiefen Einblick in die Gesprächsanalyse mit Emotionen und Dialogakten, wodurch ein nützliches Verständnis der natürlichen Sprache für eine sicherere Mensch-Roboter-Interaktion ermöglicht wird.

Der primäre Beitrag zum wissenschaftlichen Wissen der Experimente in dieser Arbeit ist das Verständnis der Auswirkungen soziolinguistischer Merkmale wie Emotion, Höflichkeit und Dialogakte, bezüglich dem Ziel, ein natürlichsprachliches Dialogsystem für eine sicherere HRI zu entwickeln. Die analytischen Exper-

imente in dieser Arbeit können als Grundlage für notwendige zukünftige Arbeiten dienen, um soziale Merkmale für die Anpassung des Verhaltens von Robotern zu integrieren. Darüber hinaus liefert diese Arbeit Wissen zur Realisierung sichererer sozialer Roboter für die Gesellschaft, mit der Fähigkeit zur verbalen Kommunikation unter Verwendung von Ansätzen aus der rechnergestützten neuronalen Linguistik, sowie um Sicherheitsbedenken zu adressieren.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Social robots and artificial general intelligent agents are expected members of the future society (Gladden, 2018). These members are expected to exhibit natural language communication, one of the fascinating capabilities humans have developed to use in daily life. While humans learn from and teach each other, mostly with verbal communication, it is reasonable to realize this existing human ability in social robots (Mavridis, 2015). It also helps to eliminate the need to require experts to communicate with the robots, and non-expert humans can naturally communicate with robots. In natural language communication, robots are expected to advance beyond the commands or instructions that can be technical and monotonous. However, it is crucial to building a language understanding model which learns conversational behaviours and nuances from the human-human interaction.

The conversation is one of the most important conventions of human communication, where the language conveys the information. Natural conversation is mostly provoked with feelings or incidences along with the information. Human communication needs to have an awareness of social cues provided through conversation by others and understand what is being spoken. The term conversation can define a casual chat as well as formal discussions. The essential processes involved in the verbal conversation are language understanding, cognitive processing and responding to the conversation partner. Different types of conversation usually govern communication and knowledge between speakers. For example, a functional conversation where some goals are to be achieved within a dialogue

with the help of information and small talks is regarded as social skills, such as greeting someone. Several factors shape the language such as grammatical syntax and structures, the cognitive knowledge of speakers, and the medium and kind of conversation (Austin, 1962; Brennan, 2000; Rashkin et al., 2018). It is essential to learn the meaning and intentions in the turns (utterances) of the dialogue for better conversational analysis, commonly with the help of dialogues acts (Austin, 1962). However, it is also crucial to investigate particular feelings behind the speaker's utterances, usually with the help of emotional expressions that help to respond with empathy (Ekman et al., 1987). Furthermore, engaging politeness of the speaker can be valuable to extend the conversational analysis and understand human behaviours, particularly during human-robot interaction.

As humans, we do not learn equally, perhaps a reason we do not react equally to the same situation (that occurs during social interaction - in conversation or on social media), as several cues and factors drive our decisions. Hence, finding a right and safer communication way becomes challenging, on the other hand, defining the right or safer situation is out of the scope of this work. However, we are fully aware that human-robot interaction certainly benefits from learning and analyzing the socio-linguistic features and behaviours in human-human interactions. Learning from different socio-linguistic features in the conversational language has some additional advantages, such as understanding dialogue initiative, multiple dialogue acts, and affective interaction; to mention a related-few from the desiderata list for human-robot verbal interaction (Mavridis, 2015). The robot has to understand a natural input language from human in all the aspects to react and follow the instructions, and eventually converse.

This thesis explores the conversational analysis and language learning for safer human-robot interaction on different aspects such as dialogue acts, emotion, and politeness. This work aims to provide a framework for human-robot verbal interaction by exclusively using socio-linguistic cues to interpret human behaviour. Understanding the human language is one of the first keys for a verbal conversation. Then the socio-linguistic cues add an interactive and significant value to produce a natural communication. We naturally learn such skills right from the early ages, for example, a spoken utterance "Could you please tell me how to reach this place on the map?" is trivial for us to comprehend and react accordingly. We can easily figure out that the above utterance represents a *question* dialogue act in a *request* form (multiple dialogue acts), which is linguistically *polite* as it

contains phrase "could you" and word "please" (Danescu-Niculescu-Mizil et al., 2013); and posses almost *neutral* emotional expression.

Another essential aspect of the dialogue is that we interpret and understand the conversation partner through the context. As social robots are on high demand to enter our daily lives, the critical feature expected is that they possess contextual inference. For example, we can reliably understand and appropriately respond based not only currently uttered sentence but also the context of previous utterances in the conversation (Bothe et al., 2018d). We propose to use attention-based recurrent neural networks (RNN) and bidirectional-RNN neural models that contextually models the conversational textual utterances to perform extensive conversational analysis, for example, using contextual recognition of dialogue acts (Bothe et al., 2018b). We perform contextual neural learning not only for dialogue acts but also for the emotion recognition using character language models to encode utterances (Bothe and Wermter, 2019). We also show how different models perform when they are ensemble together, such as RNN and convolutional neural network (CNN) models together with the word- and character-level utterance representations. It is expected that the output of speech recognition systems might contain errors, hence using an ensemble of various representations and differently behaving models becomes crucial.

In a conversation, humans use changes in a dialogue to predict undesirable and safety-critical situations and use them to react accordingly (Ekman et al., 1987). We propose to use these kinds of cues for safer human-robot interaction through early detection of dangers, especially with dialogue-based sentiment learning (Bothe et al., 2017). The socio-linguistic features, such as emotion, sentiment or politeness, together with dialogue acts, add unique value in developing a dialogue system for the robots. The robots can adapt their behaviour based on cues generated with the help of those feature recognition. We demonstrate such a human-robot verbal interaction scenario for navigating the Pepper robot that variates its speed driven by the social cues: politeness and dialogue act (Bothe et al., 2018a). We developed a dialogue system to combine these cues, which helps the robot adjust not only the navigating speed but also various social and behavioural components such as speech tone, head pitch orientation, and eye colour.

1.2 Research Questions

In our work, we focus on the natural language understanding module that helps to develop a dialogue system to adapt different robotic actions based on the socio-linguistic features found in the human input language. While developing such a system for HRI, our primary focus drives towards understanding the dialogue acts (DA) of the utterances. It is known that the application of context-based learning leads to performance gain in the task of recognition of the dialogue acts. However, we also emphasize that only current and past utterances shall be used in context for HRI scenarios that leads to the first research question:

Question 1: How can we find the number of preceding utterances in the context that are required towards recognizing the dialogue act of the given current utterance?

When this question is answered, we investigate that different dialogue acts behave differently in their context. For example, if there is an *answer* DA utterance, the previous sentences might contain a *question* DA utterance which will substantially contribute towards recognition. However, if it is a reverse case, then the contribution of the past utterances could be negligible; hence the idea is not to find any fixed number, but a generalized one that leads to the next research question:

Question 2: How much does each utterance in the context contribute towards recognizing the dialogue act of the given utterance?

Contextual behaviour in the utterances is also possible when recognizing the emotional expressions, especially in the absence of other modalities such as facial expressions or sound variations. On the other hand, the sentiment is a driver in the decision-making process. The extreme polarity sentiment utterances in the conversation are used to convey negativeness or positiveness. For example, appreciation or desirable moments are usually expressed with positive sentiment, whereas negative sentiment expresses undesirable or unhappy moments. These extreme sentiment utterances act as feedback cues as of their preceding utterances providing the context. This kind of behaviour of sentiment in the conversation leads to the next novel research question:

Question 3: How can dialogue-based neural learning estimate the sentiment of the next utterance help us find undesirable events or safety-critical cues for safe human-robot interaction?

Emotions and dialogue acts are considerably different aspects of language learning. However, the lack of availability of such a dataset that contain both the labels makes it impossible to analyze the relationships between them, that leads to the next research question:

Question 4: How can we reliably use the neural ensemble method to enrich existing emotion data with dialogue act labels? Do the emotions and dialogue acts provide any relations among themselves that would be useful to consider for conversational analysis?

In the motivation, we stated that different socio-linguistic feature for the in-depth conversational analysis could lead us to a better understanding of the human-human interaction. However, our goal is to build a language understanding module that drives to achieve social and natural human-robot verbal interaction, leads us to the next research question:

Question 5: How to combine the socio-linguistic features such as emotion or politeness with the dialogue acts in the dialogue system for HRI? How does that help to influence the output behaviour of the robots?

Our ultimate goal is to make use of the knowledge gained with these analyses and experiments for safe human-robot interaction. We attempt to discover the possibilities to utilize different socio-linguistic cues for safe human-robot interaction to increase the trust and acceptance of the robots in the emerging society.

1.3 Novelty and Contribution to Knowledge

In this work, we propose novel approaches to conversational analysis that are useful for the research community in computational linguistics and human-robot interaction.

- We propose a novel RNN-based approach on the dialogue act recognition task with domain-independent utterance representations and achieve state-of-the-art results on Switchboard Dialogue Act (SwDA) corpus. In this experiment, we use the word- and character-level language models to encode the utterances.

- The number of past utterances in the context required to recognize DA class of the current utterance is determined experimentally. We also showcase the

internal or hidden representation of the RNNs clustering the DA classes into the 2D space demonstrating the learned utterance representation possess the features isolating them in the given space.

- We developed a novel utterance-level attention mechanism configured on top of the bidirectional-RNNs to compute the contribution of the context utterances. It contains the attention mechanism that ultimately computes the weights of each utterance in the context towards recognizing the DA of the given utterance.

- We report how the context model is more reliable over no-context model predictions using their confidence values. It is clear that the context model's accuracy is consistently higher than the no-context model; however, we inspect if the confidence level of the context model is also higher.

- We develop novel ensemble models for emotion recognition in the dialogue by participating in international competitions. In the EmoInt challenge, we develop a model to compute the given sentences' intensity for classifying emotion. In the EmoContext challenge, we develop a novel model that uses the context-based ensemble of RNNs and CNNs with the word- and character-level features.

- We propose a novel approach for the sentiment-guided dialogue-based neural learning to estimate the sentiment of next upcoming utterance using RNNs. In this experiment, we show that the models learn to predict the probably undesirable or safety-critical situations that could be useful in HRI to avoid potential danger.

- We propose a novel approach to annotate emotional conversation data using an ensemble of neural annotators. We combine five different neural models (two no-context and three context models) to produce final DA classes for the given utterances. We annotate two multi-modal emotion conversational datasets IEMOCAP and MELD and make them publicly available for the research community.

- We present our discovery of unique relations between emotions and dialogue acts, and we name them emotional dialogue acts (EDAs). The EDAs show definite relations such as *Thanking* DA is mostly expressed with *Happy*

emotion and *Apology* with *Sadness*. We also investigate the failures where the model mispredicts the DA classes that help understand where the ensemble of the neural annotator fails.

- We develop a dialogue system that combines the socio-linguistic feature Politeness with dialogue acts (with added information called slot-value pairs) using hierarchical-RNN recognition models. The dialogue-based navigation HRI scenario is chosen to demonstrate the effect of change in the degree of politeness during the conversation to variate robot speed and behaviour accordingly.

- Eventually, we provide insight into language learning for safer human-robot interaction with socio-linguistic features such as emotion or politeness. We present preliminary direction for how to use those features to produce adequate and safe actions from the robots and how to integrate them into the dialogue system by navigating the robot with politeness cues.

1.4 Thesis Organization

In the first chapter, we presented motivation to this thesis work, derived research questions, and listed the novelties and contributions to knowledge. Chapter 2 provides insight into the background and conceptual methods that are used in this thesis. We briefly describe the development of natural language processing in the field of HRI and dialogue systems. We also provide a short description of what we shall expect from the experiments in this thesis. We also provide prologues on language learning by incorporating the socio-linguistic features into conversational system towards safer HRI. Chapter 3 contains an introductory background on artificial neural network methods that we use to develop our approaches. We shortly describe different RNN architectures and representation methods used in this thesis. Chapter 4 presents the approaches to recognize the dialogue acts. It contains contextual approaches based on the simple RNN and utterance-level attention-based bidirectional-RNN models with their results and conversational analysis. Chapter 5, on the other hand, provides ensemble models for contextual emotion recognition in the dialogue and sentiment-guided dialogue-based neural learning to estimate the sentiment of the next utterance in conversation. Chapter 6 presents the approach on the ensemble of neural annotators to annotate the ex-

isting emotion conversational dataset with the dialogue acts, providing a detailed analysis of the emotional dialogue acts (EDAs). It also contains the discovery of the unique relations between emotion and dialogue acts. Chapter 7 demonstrates a dialogue system for HRI where the socio-linguistic feature politeness drives navigation of the robot. It contains a method for utilizing the customized dialogue acts, in which not only intentions but also extra information is decoded from the input utterances. Finally, Chapter 8 concludes this thesis, providing discussion and conclusions on the experiments and results conducted in this thesis work. It also contains answers to the research questions posed in this first chapter, together with the possible future work.

Chapter 2

Language Learning for Human-Robot Interaction

This chapter focuses on the methods to incorporate techniques of natural language processing for conversational analysis and human-robot interaction (HRI). Humans will be able to interact naturally and verbally with robots is still futuristic. However, there is plenty of research work proven the early steps towards a robust conversational HRI. In this chapter, we will discuss the most important and relevant developments in this regards.

2.1 Introduction

Robots with the natural-language conversational ability make them useful in direct human-robot interaction applications, such as health, education or retail. However, being in the direct interaction with humans, safety comes first, for which understanding different aspects of the conversation becomes crucial, for example, conversational and discourse analysis, contextual and pragmatic behaviours in conversation, and affective or emotional comprehension. When a person wants to give a command or order the robot, verbal interaction could make them feel natural (depicted in Figure 2.1) than conveying commands as technical terms or from a graphical user interface (GUI) of the smartphone. Moreover, most of the conversational robots are not directly equipped with learning capabilities (Mavridis, 2015), and they are usually task-oriented human-robot interaction scenarios (Steinfeld et al., 2006; Bothe, 2015). However, it is still necessary to understand why is it essential to have robots with natural-language capability,

Figure 2.1: Human-robot Interaction Scenario.

and what should we expect from the conversational robots?

Many works have attempted to create a unified set of requirements for the conversational HRI (Steinfeld et al., 2006; Pandey, 2012; Mavridis, 2015). Some of the primary expected abilities are grounding speech acts, affective interaction, dialogue initiatives and learning from human conversations. This thesis attempts to understand how to model and analyze the abilities mentioned earlier to deploy them in HRI applications. We propose to use the presented methodologies to learn and analyze such linguistic aspects using deep learning techniques. It is crucial to understand that these abilities are not limited and could be extended further, such as multiple speech acts, multi-level learning, mixed-initiative dialogues, along with the utilization of online resources and services (Mavridis, 2015). Moreover, to build a conversational system for HRI includes a different perspective than human-computer interaction (HCI) systems. Apart from any damage from the HCI system, HRI has to follow the fundamental Three Laws of Robotics as defined by Isaac Asimov (Asimov, 1963).

Hence, we derive motivation for this thesis to explore various socio-cognitive and -linguistic building blocks, having safety concerns in the first place. We aim to bring conversational abilities, such as context- and situation-aware response, affective behaviours and proactivity, to the robotic scenario through conversational analysis. We also highlight social norms for dialogue, comprehension and navigation to reduce efforts and confusion in the given scenario. Our main aim is

to learn from the datasets of our day to day conversational activities, and finally, design and develop algorithms and frameworks to equip the robots with such abilities.

2.2 Language to Verbal Interaction

As a human being, the ability to have a conversation represents a crucial cognitive component of social skills (Riggio, 1986). The humans can reliably understand each other and communicate verbally and non-verbally, where "verbal interaction is the basic reality of language" (Allen, 1993). Language has been a subject of study from ancient linguist Panini in sixth century BC, through the philosophers like Plato and Aristotle, and then to the 20th century's most influential linguists like, to mention a few, John Austin, John Searle and Naom Chomsky (Rajagopalan, 2000; Bod, 2013; Kadvany, 2016). Language and communication and how it works have been a point of debate for a long time. However, one of the ideas the debate converged to, is that language is not just the symbols, words, sentences or grammar, but it is their production and issuance in the performance of the speech acts, as defined by Searle (Searle and Searle, 1969; Rajagopalan, 2000). Hence, language is not only the combinatorial possibilities of the symbols to make the well-formed sentences, as a Chomskyan generative grammarian would claim, but the contextual knowledge the speaker has in the conversation.

In recent decades, computational techniques have taken over the traditional ones in linguistics. It was possible due to recent advances in artificial intelligence and data-driven modelling in the machine learning field. The primary attention has been driven towards artificial neural networks, and its prominent extension called deep learning. All these advancements brought us to numerous possibilities for natural language processing. They enabled us to build the conversational dialogue systems, most importantly, neural conversational agents with the help of dialogue corpora (Serban et al., 2015; Gao et al., 2018). Natural Language Processing (NLP) aims at converting natural human language into computer representations, i.e. symbolic or numeric representations that are easy to handle for computers. NLP involves several challenging tasks such as natural language understanding, part-of-speech tagging, language modelling, natural language generation, automatic summarization, sentiment analysis, and discourse analysis. When combined to architect a system, these tasks enable building an ef-

fective conversational dialogue system for human-computer or -robot interaction. In the following section, we will briefly review the conversational systems, a typical dialogue system and how NLP tasks can be integrated to build a particular application, and we will also discuss their use in HRI applications.

2.3 Natural Language Processing (NLP) for Conversational Systems

The conversational systems can be grouped into three categories: question answering system, task-oriented dialogue system, and chatbots (Allen et al., 2001; Gao et al., 2018). Question answering systems are usually designed to directly answer the questions based on rich knowledge, like asking about the weather forecast. Task-oriented dialogue systems are the most widely accepted architectures; they are modular and provide substantial opportunity to improve each of the components, this approach is commonly used in HRI scenarios. On the other hand, the chatbots are usually developed to perform small talks such as "tell me a joke" or greetings, and usually, they are trained as data-driven models.

Most of the systems fall under the category called spoken dialogue system as the input and output are bound with speech interface. A typical dialogue system, as shown in Figure 2.2, is composed of four primary modules: Natural Language Understanding, Dialogue Manager, Response Manager, and Natural Language Generator. A Natural Language Understanding (NLU) module identifies user intentions and extracts associated information from the input utterance. A Dialogue Manager (DM) keeps track of the dialogue state that captures all essential information in the conversation and may communicate with other task-oriented databases as if needed from the interpretation of NLU. DM module is usually responsible for communicating with databases depending on the task to be accomplished for a particular goal, like asking about the restaurants in the city. A Response Manager (RM) is a DM dependent module that takes care of the kind of response generated and usually waits in a loop with the DM for continuous corrections. Natural Language Generation (NLG) module is responsible for converting agent actions from the RM to natural language responses. The input and output are accomplished with the help of automatic speech recognition and text-to-speech synthesis, respectively.

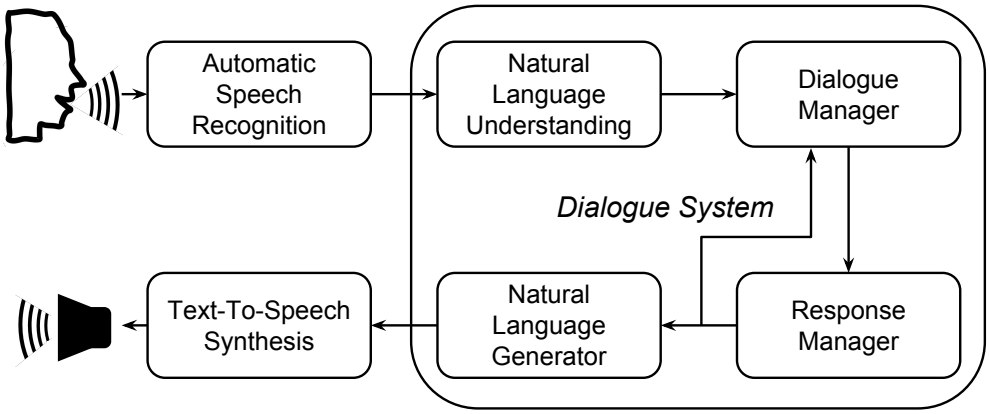


Figure 2.2: Typical Dialogue System.

While being a modular architecture, it provides substantial opportunities to improve each component in the dialogue system, and it also allows to control them independently. Such flexibility is useful for developing the dialogue systems for human-robot interaction where the system could be modified to add different linguistic features and modalities (Shi and Yu, 2018; Bothe et al., 2018a). Such a system is proposed and presented in Chapter 7, where the responses and robot behaviour are modulated with two linguistic features, the dialogue act (intention) and politeness.

Recently, there have been several attempts to develop entirely data-driven systems, popularly called end-to-end conversational models (Gao et al., 2018; Ritter et al., 2011; Vinyals and Le, 2015; Weston, 2016). The machine translation techniques mostly inspire the end-to-end conversational approaches (Kalchbrenner and Blunsom, 2013a; Sutskever et al., 2014; Yang et al., 2017) where a deep sequence-to-sequence neural network directly maps the user input to the conversational agent output. They are gaining popularity due to ease of training on the big data in an unsupervised fashion. For example, sequence-to-sequence model could be trained on a large number of conversations such as movie subtitles, where utterances after utterances are trained as input and output sequences (Vinyals and Le, 2015). The conversation with such agents turns out quite random as any input utterance gets mapped to individual responses or a combination of the words to form an output utterance from the learned data. However, to mitigate such phenomenon, one more kind of dialogue modelling is gaining popularity called goal-oriented end-to-end models (Hori and Hori, 2017; Ultes et al., 2017; Lu et al., 2019). In this case, the conversational agent's goal is to respond on a

particular domain given a history of dialogue, for example, recommending places to visit, suggesting restaurants in the city.

In our experiments, we mostly use the previous version of the dialogue system, which maps input utterances to responses with the modular components. Our primary focus of research is on the natural language understanding module of the dialogue system, and hence in the next sections, we will depict some of its essential aspects. The neural techniques used for language processing will be explored in Chapter 3 Neural Networks for Natural Language Processing.

2.4 Natural Language Understanding (NLU) for HRI

Natural Language Understanding (NLU) is a crucial process in the dialogue system and a challenging natural language processing task. NLU is also popular due to its commercial use in a variety of applications such as text categorization (dialogue act and intention recognition, sentiment analysis, emotion analysis), automated reasoning (semantic parsing and analysis), machine translation, question answering, news-gathering, and large-scale content analysis (Macherey et al., 2001; Hirschman and Gaizauskas, 2001; Van Harmelen et al., 2008; Fernández-Martínez et al., 2012). As a general overview shown in Figure 2.3, NLU processes sit in the core of NLP tasks. Contrary to human-level language interface, which is mostly speech, the NLP tasks are solved at text level and then interlinked with automatic speech recognition (ASR) and text-to-speech (TTS) synthesizer. The ASR task is also considered as a part of NLP; however, ASR takes an acoustic signal as an input and returns a word graph hypothesis. NLP then takes such an output data stream from ASR and extracts meaningful representation through NLU, as depicted in Figure 2.2 of the dialogue system. As we have already mentioned in Section 1.3, this thesis's main contributions lie in the field of natural language understanding.

For language understanding, as mentioned, the first step is automatic speech recognition. The ASR enables the system to listen to a person and convert the spoken speech into text. This text is further processed for the language processing modules like NLU for either dialogue act recognition, intention detection, sentiment or emotion analysis. Understanding the spoken language is much more

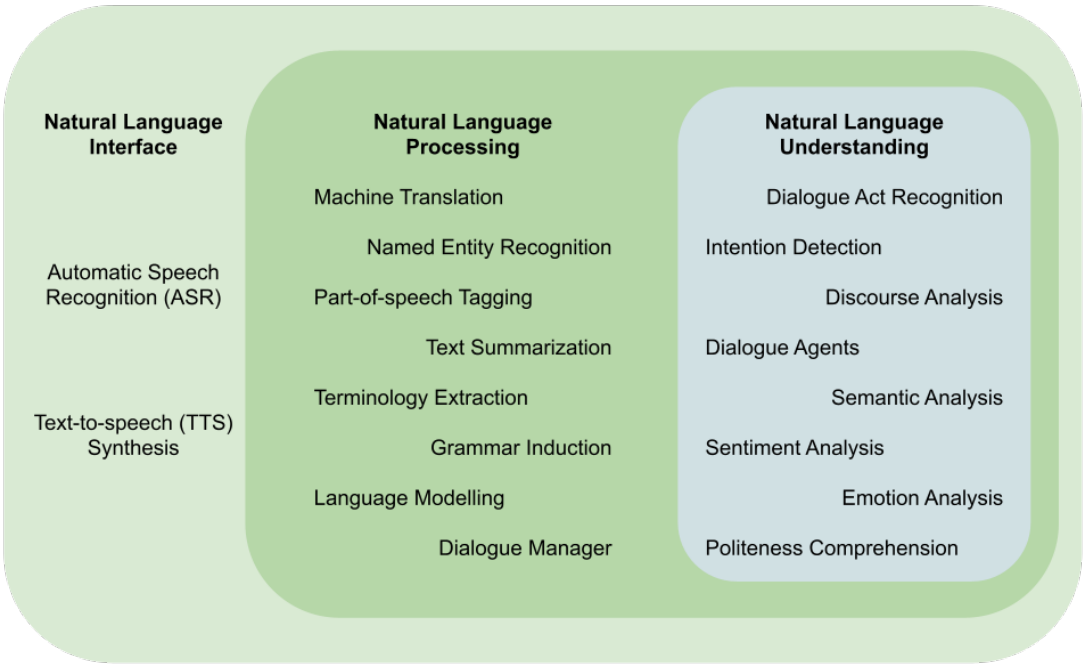


Figure 2.3: NLU Research in NLP (an overview).

diverse than ASR, as ASR has a specific and straightforward task which is converting speech into text. For example, in the scenario of NLU where intention detection in terms of question and answers are essential while in other scenario detecting whether the input spoken utterances are commands or not. There can be many ways to interpret the text or input utterance, which diversifies the desired output from the NLU module. Interpreting the meaning may require to identify some keywords available in an input utterance while in other cases, decoding a piece of in-depth semantic information might be crucial. In short, developing a complete language understanding module is out of the scope of this thesis work. However, developing the modules that can constitute a meaningful NLU for HRI is always possible for the given scenarios.

The methods from machine learning technology have been successfully used for several NLP tasks, for example, the use of Principal Component Analysis (PCA) and Singular Value Decomposition (SVD) algorithms for finding the word embedding. The word co-occurrence is used as a parameter to learn the position of the surrounding words (Lebret and Collobert, 2013; Glorot et al., 2011). Since learning approaches began to overtake traditional methodologies, the words are being represented with vectors providing an ability to handle them in matrix calculus (Bergman and Davidson, 2005; Mikolov et al., 2013a). Many NLP tasks

traditionally treated that way, for example, part-of-speech (POS) tags such as nouns (N), verbs (V), and subjects (S) being represented in discrete categories which are being replaced by vector representations.

Recently, the deep learning approaches allowed to encode language features into the vector representations and structure the relations between words and phrases with rich information. The deep learning approach, like the recurrent neural network (RNN), is sometimes jointly used with traditional approaches to achieving rich semantic information. For example, recently, context-free grammar (CFG) was used jointly with RNN, where CFG recognizes syntactic structures and RNN finds compositional semantic relations (Socher et al., 2013a). The ability to capture such semantic information against syntactic structure has benefits of resolving the ambiguous sentences. For example, `"go to the right"` provides information about the "direction to take" as against to `"go to the kitchen"` gives information about "where to go" (naming the place). Enthusiastic reader may jump to Chapter 7 to find the use case of such semantic decoding in Table 7.1, where first sentence could be decoded as an intention *MoveRobot* with the *direction* to *right* while second sentence *TakeToPlace* with the *room* name *kitchen*.

The interpretation of the input utterance can be perceived with different features. For example, the social service robots are supposed to be in the daily human contact with verbal communication; in such a case, it is useful if they learn and understand the socio-linguistic behaviours. Eventually, learning to avoid undesirable or potentially dangerous situations in human-robot interaction scenarios such as human conveying the message that the glass being used is broken, which robot could understand by using dialogue-based sentiment learning (Bothe et al., 2017). On the occurrence of the input utterance from human `"Wait, that glass seems broken."` robot could understand the negative sentiment in the dialogue context, as depicted in Figure 2.5 discussed in Section 2.4.3. The robot could raise the question of whether to continue and hence potentially avoid dangerous action. Also, in the previous examples, `"go to the right"` and `"go to the kitchen"`, if we combine the understanding of politeness comprehension, the robot can achieve the ability to know if the person is in a hurry or patient. For example, if the person says `"Could you please go to the kitchen?"` instead, linguistically it is a polite sentence and does not show any directive command to robot. In such a case, the robot could politely respond and take appropriate actions (Bothe et al., 2018a).

In the following sections, we discuss such socio-linguistic features; those can be mutually exclusive and are applied at different interpretation stages as per requirement.

2.4.1 Dialogue Act Recognition

In linguistics, the dialogue act represents a performative function of an utterance, for example, the utterance might be a question, a statement, an answer, or a request (Stolcke et al., 2000). For instance, the sentence `"Could you show me the kitchen?"` can be defined as a *question* (more precisely a yes-no type or a request). In particular to natural language understanding, the dialogue act plays an important role in the context of conversational dialogue learning. It is a commonly used linguistics feature for conversational and discourse analysis to quantify and identify the role of utterances (Grosz, 1982). The recent use of dialogue acts can be found in many applications, such as conversational dialogue systems (McTear et al., 2016).

The research on dialogue act recognition has increased since its successful use in spoken dialogue systems (McTear, 2002). We mainly focus on the dialogue act recognition as it is one of the core tasks in NLU. The traditional machine learning and statistical approaches were used to recognize the dialogue act, such as the Hidden Markov model, to classify the utterance (Wermter and Löchel, 1996; Stolcke et al., 2000). Artificial neural networks have recently been successfully deployed to recognize and classify the dialogue acts (Kalchbrenner and Blunsom, 2013b). However, modelling the dialogue acts at an utterance level drops the contextual information coming from the preceding utterances. Hence, new modelling techniques have emerged where context-based neural architectures are used to achieve the same task (Kumar et al., 2018; Chen et al., 2018b; Bothe et al., 2018d).

A conversational system typically consists of a taxonomy of dialogues that specify different functions of the utterances. These functions includes different actions, for example, in question-answering dialogue system the actions would be *question* and *answer*. There have been many taxonomies, most popular *speech acts* (Austin, 1962), which forms a basis for many further studies. That was later modified into five classes (Assertive, Directive, Commissive, Expressive, Declarative) (Searle, 1979). Then new taxonomy emerged which is very fine-grained,

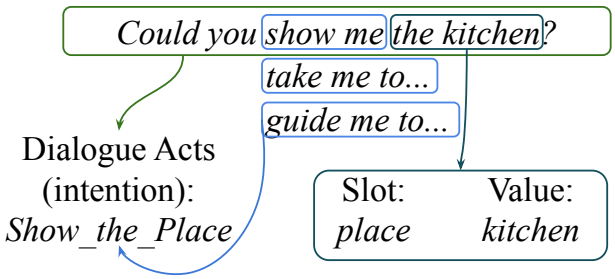


Figure 2.4: Example on Semantic Decoding.

including 42 dialogue acts, called the Dialogue Act Markup in Several Layers (DAMSL) tag set. In this taxonomy, each DA has a forward-looking function (such as Statement, Info-request, Thanking, Question) and a backwards-looking function (such as Accept, Reject, Answer) (Allen and Core, 1997). We will see more details on this in Chapter 4, where we also model the dialogue act learning with neural models, especially RNNs, more details in Section 4.4 (Bothe et al., 2018d). We have extended this experiment in Section 4.5 where we can compute the amount of contribution of the preceding utterances in the context using Bidirectional RNNs with attentive neural models (Bothe et al., 2018b).

2.4.2 Semantic Decoding for Dialogue Systems

The dialogue act recognition provides a sufficient amount of understanding aspects of the language and commonly used for conversational analysis. However, it might not be sufficient for a dialogue system to formulate the response only with such information about the input utterance like a *question*, an *answer*, or a *request*. Often, when we listen to the utterance, we try to extract as much information as possible from the sentence. For example, the utterance `"Could you show me the kitchen?"`, we could found that it is a *question* with an intention to show or take to someplace. However, one has to extract that extra information along with *show* something and it is the *kitchen* which is a *room*, see Figure 2.4, which is necessary for the HRI dialogue systems.

Semantic decoding can provide a framework to extract such a piece of information. Traditionally, semantic grammar and rules are used to classify the parts of the utterance in terms of semantic roles. The task is to detect semantic arguments associated with the verb as a predicate of the sentence and nouns as the agents and themes. For example, in the sentence `"He is showing the kitchen to John."`,

show is the main verb so that constitutes a predicate, *he* would be an agent who is showing, *the kitchen* forms a theme, and *John* is a viewer. It is also known as a slot filling or semantic role labelling task, and deep learning techniques have been successfully deployed to solve this problem, specifically recurrent neural networks (Mesnil et al., 2013). The common uses of the semantic decoding are in a domain such as flight reservations, hotel and restaurant recommendation systems. The statistical modelling methods such as Conditional Random Fields (CRFs) have had great success in this task, particularly on the Airline Travel Information Service (ATIS) benchmark. However, RNN-based models outperformed the CRF baseline, improving the error reduction (Mesnil et al., 2015). The task was to recognize the intention (extended but domain-specific dialogue act) and also fill the slots. For example, the sentence `"search the flights from Hamburg to Paris today"` has an intention *find_flight* and three slots. Hamburg forms a first slot *departure_city*, Paris a second slot *arrival_city* and third slot is *date* by giving information such as 'today'.

We find some similar frameworks used in the domain of the robotics instruction decoding (Fong et al., 2003b). For example, `"go near the table in the kitchen"`, from HuRIC corpus, task is to classify the intention as *going* with the *Agent* being a *robot*, *Theme* would be *table* and *Goal* would be *kitchen* (Bastianelli et al., 2014). Similarly in Tell Me Dave corpus, the higher level of instructions as intentions are converted into a set of symbolic actions (Misra et al., 2016). For example, `"Put the mug into the microwave"` has intention *Boiling the Water*, where system might need to decode this instruction into *Move-to Mug*, *Grasp Mug*, *Move-to Microwave*, *Open Microwave* and then *Put Mug in Microwave*. A reinforcement learning approach could also be used to generate the sequence of actions from the set of symbolic actions (Zamani et al., 2018). The utterances could formulate a meaningful structure such that the information could be used to accomplish the dialogue system's input-output cycle. One of the popular dialogue systems, called PyDial (Ultes et al., 2017), uses similar semantic decoding framework. The input utterance is structured with slot-value pairs along with intention. In our example utterance `"Could you show me the kitchen?"`, the intention can be seen as *Show_the_Place*, and {*slot: value*} pair would be {*place: kitchen*} (Bothe et al., 2018a). We will explore such examples in Chapter 7 Section 7.3.1 and their use in human-robot interaction scenario.

2.4.3 Sentiment and Emotion Analysis in Dialogue

The sentiment is an essential characteristic feature in the decision-making process and thus has received much attention in socio-linguistic studies (Pang and Lee, 2008). The sentiment or emotion has also been considered as one of the primary social cues in conversational analysis (Vanzo et al., 2014; Bothe et al., 2017; Gupta et al., 2017; Shi and Yu, 2018). Sentiment can be seen as a grounded emotion, whether a spoken or written sentence has positive or negative polarity. Emotional intelligence for sentiment analysis has recently introduced several computational linguistics tasks such as natural language processing and text mining (Fischer and Steiger, 2020). Such a study provides deep insight into the affective states and subjective information of the emotions (Bothe and Wermter, 2019; Bothe et al., 2020). Sentiment analysis is often used in understanding customer reviews; for example, how they like certain products against others. It is also applied in the field of healthcare matters to assist patients with better service (Yadav et al., 2018).

In the following sentences, sentence (1) can be perceived as positive against sentence (2). In some cases, the intensifiers can be used to express the sentiment or emotion with a higher degree. As given in the sentence (3) bellow, "so" is used to intensify the positive sentiment, similarly "very" or "too" words can be used to intensify the emotions (Lakomkin et al., 2017; Mohammad and Bravo-Marquez, 2017b). We have conducted such an experiment in Chapter 5 Section 5.2 Emotion Intensity Detection from the Sentences, where the particular emotion is classified with fine-grained intensity values using the combination of traditional and neural networks.

(1) I am happy for you.

(2) Feeling worthless as always.

(3) I'm just still. So happy.

It is crucial to clearly define the sentiment or emotion classes for the annotation purpose so that they can be interpreted in the same way. Such a scheme might loose interpretability and thus often the complete emotion classes are used to represent the sentences, for example, *happy*, *sad*, *angry*, etc. (Mohammad and Bravo-Marquez, 2017a; Sailunaz et al., 2018). However, the problem remains the same when it comes to the ambiguous sentences, for example, Why don't you ever text me? or Me too! It is difficult even for humans to

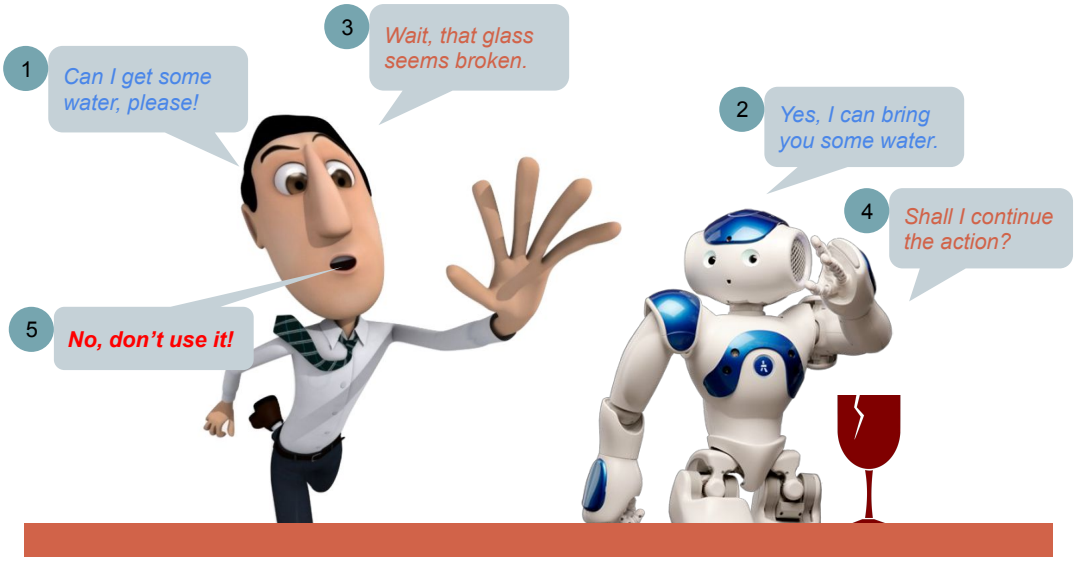


Figure 2.5: Illustrating dialogue scenario to potentially avoid a dangerous action using dialogue-based sentiment learning. Numbers are referring to the sequence of utterances in the conversation.

identify the emotion of such sentences unless the context is given. One might say that the above sentence `Why don't you ever text me?` is angry or sad. Nevertheless, the sentence `Me too!` is far from imagining any of the emotions. Hence contextual information in the text is essential, and such a context could come in the dialogues (Bothe and Wermter, 2019). We conducted such an experiment with the contextual emotion detection in dialogues in Section 5.3. The contextual information from the preceding utterances is used to recognize the emotion of the utterance with the help of the ensemble of several neural models.

As illustrated in Figure 2.5, when a human in a conversation gives some cues related to safety, those cues could be recorded and potentially used for the human-robot interaction scenario. The learning approach could be seen as teacher-student learning through feedback (Latham, 1997) and could be potentially applied for the conversational analysis and dialogue-based learning (Weston, 2016; Bothe et al., 2017). We experimented with such dialogue-based learning in Section 5.4, where we use the recurrent neural networks to model the learning process. The dialogues were modelled in such a way that a set of utterances forms the context and given this context task is to learn the sentiment of the upcoming utterance. This way, we achieve the learning through feedback as the positive or

negative sentiment allow to adjust the learned weights of the neural model. In this experiment, we use the sentiment feedback of the extreme polarity of the utterances such as `"No, don't use it!"` as depicted in illustration of Figure 2.5. On the other hand, eventually, the models also learn to predict the sentiment of the next utterance. Our experiments on this study can be visited in Section 5.4. The emotions and dialogue acts possess unique relations that are explored in Chapter 6, under the title Emotional Dialogue Acts (EDAs).

2.4.4 Politeness Comprehension in Conversation

Politeness is considered as a feature of representing good manners. It also gives an implicating effect in the conversation, such as how much to say. For example, negative politeness signifies 'do not say more than is necessary' whereas positive politeness signifies 'say as much as required' (Brown and Levinson, 1987; Watts, 2003). In our study, we might not focus on the counterpart to politeness, i.e. rudeness. However, we would like to stress on the linguistic issuance of the politeness. When saying `"Get me some water."` against to `"Could you please get me some water?"`, we can see that the first statement is direct while the second version of the utterance is featured with politeness. The words such as *please* and *could you* effectively puts much more weight to indicate politeness (Kasper, 1990; Aubakirova and Bansal, 2016). It depends on the kind of discourse posed by the speaker, such as demand, request, suggestion, or hint. In conversational discourse analysis, socio-linguists posited and demonstrated that the style of discourse type produces constraints on speakers' linguistic behaviour (Saville-Troike, 2008). However, it must be kept in mind that the discourse analysis is non-objective study, but it is a useful tool for comprehending politeness in the social conversation.

We demonstrated with an example of the politeness detection in Chapter 7 and its integration with dialogue acts for the human-robot interaction scenario (Bothe et al., 2018a). The dialogue act or intention of the sentences might remain the same while the politeness or the level of politeness might differ a lot. For example, in our last examples, `"Get me some water."` and `"Could you please get me some water?"`, the intention is same *ordering water* however, first utterance is like a *demand* while other is more like a *request*. In our scenario, when we order or command to the robot, the necessary step is to

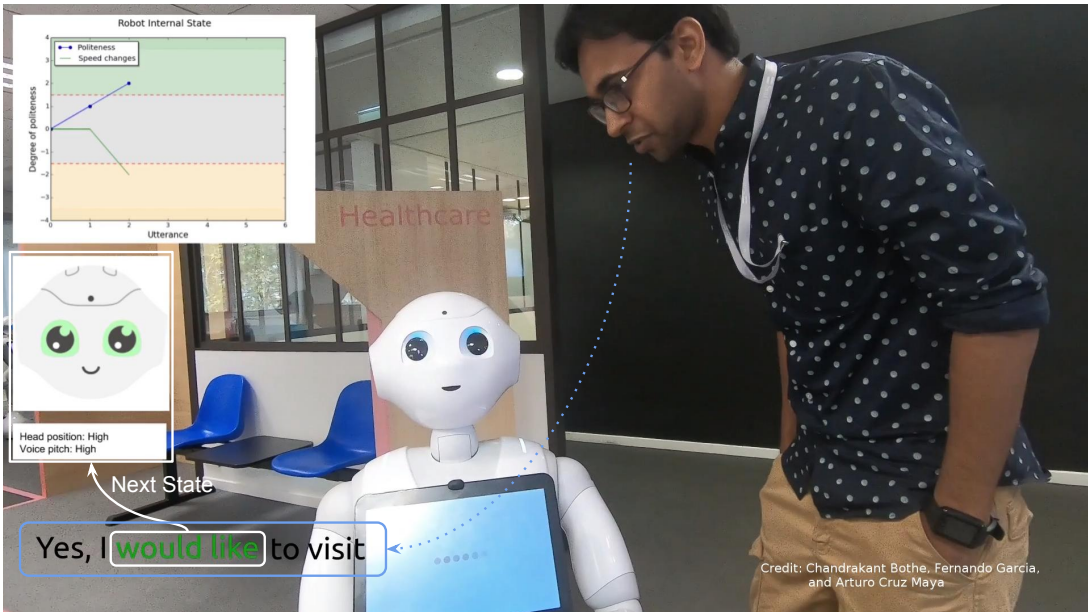


Figure 2.6: Illustrating a linguistically polite dialogue scenario.

detect if the input utterance is polite or impolite to understand the more in-depth discourse such as request or demand. It could help the robot to comprehend the socio-linguistic behaviour of human for taking appropriate decisions and actions. For example, being impolite may intend that the user is in a hurry and as in such a case the sentences are usually shorter. In such a situation, we want the robot to act quickly than talking about the things that might not be necessary. On the other hand, such a piece of information is also helpful to variate multiple robot behaviours during the interaction, as illustrated in Figure 2.6.

2.4.5 Conversational Analysis

Interactive communication between two or more people forms a conversation where different media could be used, such as spoken, written or sign language. In the spoken language talk, speakers use utterances in their turns. The conversational analysis attempts to explain how or why someone would utter a particular utterance centred on tasks or institutions (Wooffitt, 2005; Gibbs and Van Orden, 2012). For example, conversation occurring in politics, courts, helplines, and educational settings. The conversational analysis is used in many fields, with minor variations and adaptations, with one of the most successful and distinctive approaches to analyzing the socio-linguistic interactions. Discourse analysis is

mostly confused with the conversational analysis approaches; however, discourse analysis acts on a broader level to comprehend the consequence of the sequence of the turns in conversation.

On the one hand, researchers attempt to analyze and explain the conversational phenomenon using these approaches. On the other hand, the ability to create an artificial conversational agent that cannot be distinguished from a human participant remains a test of complete artificial intelligence (for example The Turing Test). We will discuss these analytical techniques of the conversation in the following sections and also look into their potential use for conversational language learning. Beyond language understanding or comprehension approach (in Section 2.4) can be achieved from the conversational analysis. The context-based dialogue act recognition and dialogue-based sentiment learning are examples of such learning approaches.

Pragmatics and Hierarchies in Conversation We could already see how the context plays a vital role in conversational analysis, and for the tasks such as dialogue act recognition or emotion and sentiment analysis in dialogues. Pragmatics is the study of context contributing to the meaning of the current situation or an utterance in the conversation. How do people decide how to respond in context? What is the information being used when answering the question? Is it dependent on the knowledge of a speaker or individuals understanding of the situation? Many such factors affect the meaningfulness of the spoken utterances in conversation. Mostly in pragmatics theory, it is assumed that people have a particular knowledge of the situation to utter certain words and fails to both the regularity and variability in peoples speech behaviours (Gibbs and Van Orden, 2012).

Discourse versus Conversational Analysis Discourse and conversation analysis has many similarities; however, they are different in some aspects. Conversation analysis uses everyday natural language to analyze how we perform interpersonal actions and how we use them to interact socially. On the other hand, discourse analysis treats language on a broader level and looks for the consequences that might be affecting a sequential context in the conversation. Both of the analysis processes are qualitative in nature and analyze the functional importance of utterances and fundamental properties of the language (Wooffitt,

2005).

Moreover, the discourse analysis approaches are applied to written, spoken, or sign languages. It is widely used in various fields such as social sciences, psychology, politics and many others, including linguistics. In this thesis, we mostly focus on the spoken language in the form of text. We analyze the utterances for different dialogue acts (explained in Section 2.4.1). We emphasize more on the context-based learning achieved using neural networks. The preceding utterances contribute to the recognition of dialogue act of the current utterance (Bothe et al., 2018d) and we have created a web demonstration[1](Bothe et al., 2018c). Further, we investigate the preceding utterances' contribution to the current one by using the attention-based neural model (Bothe et al., 2018b).

2.5 Conversational HRI for Social Robotics

Our ultimate goal is to demonstrate a conversational dialogue system for social robots that can incorporate the socio-linguistic cues to adapt to social behaviours. Autonomous adaptation to the social behaviours based on such cues brings robots to a higher degree of decision making autonomy. The studies found that the highly autonomous robot influences more on human decisions than a lowly autonomous robot (Rau et al., 2013). As a result, it also provides additional value to trust and acceptance of the robots in society. The levels of autonomy in human-robot interaction are listed for the reference in Table 2.1, it is based on (Sheridan and Verplank, 1978) from (Rau et al., 2013). It is important to note that for the conversational social robots to achieve the highest level of autonomy, the language understanding process has to be very robust (Beer et al., 2014).

The conversational system for social robots needs a grounded analysis of the human language that follow behavioural psychology. For example, understanding the politeness strategies for the HRI as humans do and then apply the same to the robots (Bothe et al., 2018a). When we are in an urgent situation, we use short utterances instead of thinking of etiquette or social norms. As a result, it makes the utterances linguistically impolite, for example, asking `"Get me some water."` instead of `"Can I get some water, please?"`. In the first case, the utterances sound like an *order*, and in the second case, it sounds like a *request*. In both cases, the robot decides by itself (autonomously) how would it react to the given input

[1]Discourse Wizard: `https://crbothe.github.io/discourse-wizard/`

Level	Robot Actions
(1)	Robot offers no assistance; human does it all.
(2)	Robot offers a complete set of action alternatives.
(3)	Robot narrows the selection down to a few choices.
(4)	Robot suggests a single action.
(5)	Robot executes that action if human approves.
(6)	Robot allows the human a limited time to veto before automatic execution.
(7)	Robot executes automatically then necessarily informs the human.
(8)	Robot informs human after automatic execution only if a human asks.
(9)	Robot informs human after automatic execution only if it decides to.
(10)	Robot decides everything and acts autonomously, ignoring the human.

Table 2.1: Levels of autonomy for human-robot interaction, source (Rau et al., 2013).

utterance issued from the human user.

2.6 Towards Safe HRI using Language Learning

During the human-robot interaction, a conversational agent needs to keep track of the human user's input utterances. The behavioural changes do not occur only in the instance of one turn but from the history and context of the conversation. It also varies given the knowledge of the speaker partner, for example, if we meet a new person perhaps we try to follow certain etiquette, and if we know the person from a long time, one may choose an informal language. It also depends on the professional hierarchies and relations (Langlotz and Locher, 2017). However, when it comes to robots, we want them always to be polite but modify a certain level of politeness depending on the urgency, as explained in Section 2.4.4. Hence, in HRI, it becomes essential to maintain the history of the user input utterances and infer the behaviour from the context.

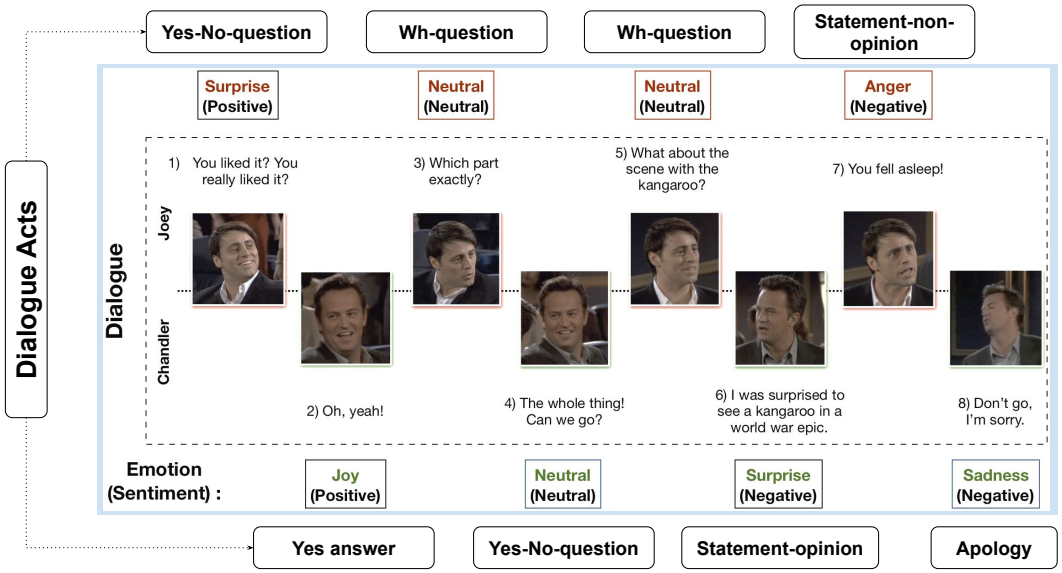


Figure 2.7: Example of the dialogue where conversational behaviour changes suddenly, from the emotional dialogue acts Bothe et al. (2020), we add dialogue act labels to the original image from Poria et al. (2019).

For example, see in Figure 2.7, the talk between Chandler and Joey, from utterance number 1 till 6 all the conversation is positive or neutral (Poria et al., 2019). However, suddenly when Joey realizes that Chandler did not pay attention, and he got angry. Chandler responds to Joey with Apology and Sadness instead of getting back angry. Linguistically, he is trying to play safe and avoid a potentially undesirable situation that could occur, so as not to make Joey unhappy. Chandler has selected a desirable, favourable and polite action, following the social norm and etiquette. The etiquette comes from the long term engagement with the speaker. On the other hand, such changes in the conversation cues might provide feedback to learn, as explained in Section 2.4.3.

Hence, learning desirable or safety-critical situations from the language becomes possible for safe HRI. For such safe HRI scenarios, the natural language understanding module should be able to learn about all possible socio-linguistic features. As shown in Figure 2.8, the utterance is decoded into the information of multiple features. As shown, the dialogue acts could be decoded into multiple levels, such as it is a *Yes-No Question*, and at a lower level, it is a *Request*. Specifically for HRI scenarios, the dialogue acts have to be customized and extract the

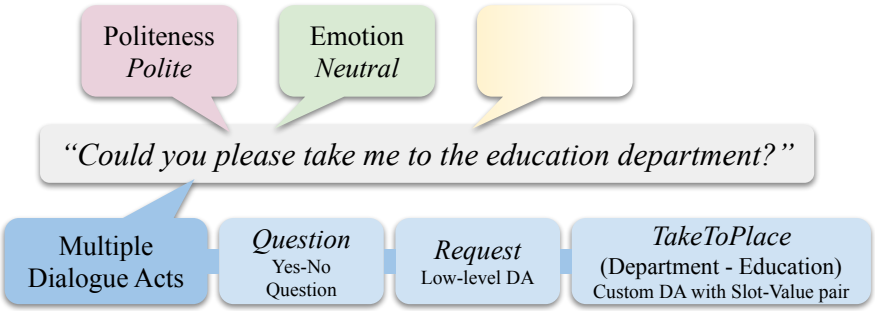


Figure 2.8: Decoding multiple socio-linguistic features.

extra information, such as in this case, the user is asking to take at the education department. NLU also needs to decode features like politeness and emotion. This way, the robot would be able to learn and understand humans better than using only dialogue acts, as in the traditional dialogue systems.

2.7 Summary

In this chapter, we discussed the background on the methodologies proposed in the thesis. We briefly describe the research background in natural language processing and dialogue systems. We provide an insight into the proposed methods and necessary concepts towards language learning for the safe human-robot interaction. We also present a brief introduction to some socio-linguistic features such as dialogue acts, emotions or sentiment and politeness.

Chapter 3

Foundation of Neural Networks for NLP and HRI

In this chapter, we will discuss and learn about the techniques in artificial intelligence that are used for natural language processing and human-robot interaction in the experiments of this thesis. We also explore language representations such as word embeddings and language models that are often deployed to represent the natural language input to the neural models.

3.1 Introduction

Artificial neural network (ANN) is a computing framework loosely based on biological neural circuits of the animal brain (van Gerven and Bohte, 2018). The ANN can be seen as a network or circuit of neurons or nodes, used as a solution for artificial intelligence (AI) problems. Similar to the biological neural circuit, ANN neurons are modelled as weights, a positive weight shows as an excitatory connection link, while a negative weight representing inhibitory connections. The history of neuronal learning traces back in the late 1940s, when a learning hypothesis based on the mechanism of neural plasticity was designed by D. O. Hebb (Hebb, 1949), which later popularly became known as Hebbian learning. Contemporary comparison of a biological and artificial neuron is presented in Figure 3.1. The output is achieved by the weighted sum of the input and connecting weights as a linear combination. An activation function is used to control the output amplitude, usually in the range of 0 and 1. The main idea behind the ANN approach was to mimic the human brain processes, so that machine can

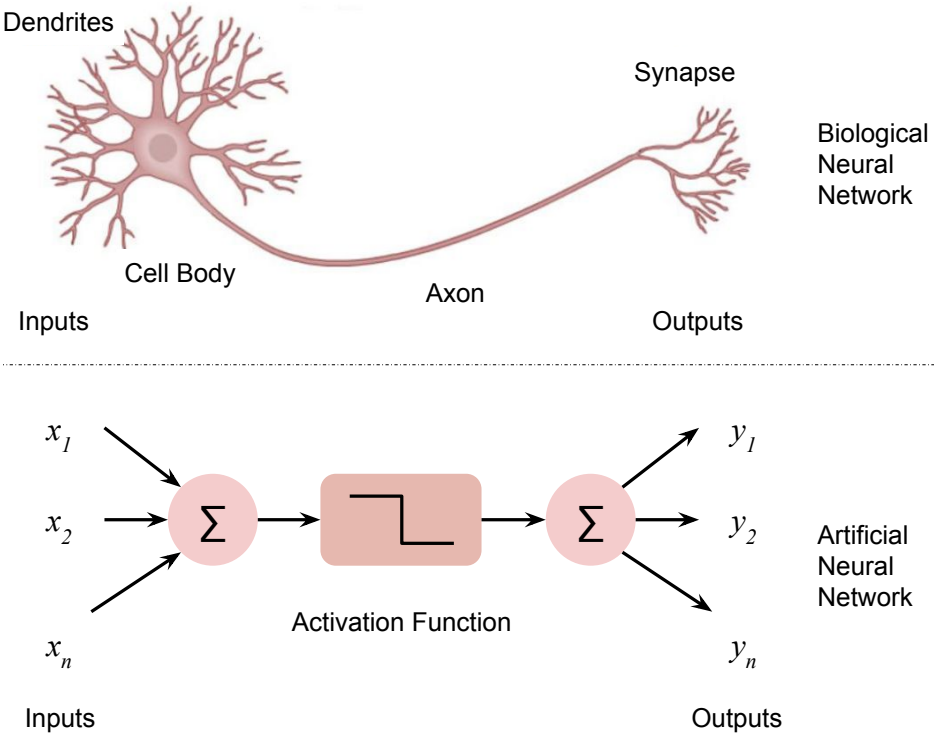


Figure 3.1: Biological vs. Artificial Neural Networks.

learn the way humans learn. However, eventually, the attention got deviated from biology while leading to success in numerical computations. One artificial neural network might be composed of several such neurons to form a particular network or circuitry. The common practice is to use these artificial neural networks for predictive modelling where networks can be trained on a particular dataset. These networks are capable of learning from the input features directly without having prior knowledge of the input data.

ANNs are successful in solving specific problems such as speech recognition, machine translation, computer vision, email spam filtering, playing games, and pattern recognition. For example, input text (encoded into vector representation) is labelled with sentiment values, such as positive and negative. The network can be trained directly with the given labels for the sentiment analysis task. The trained network might learn to identify the sentiment of the input text without being known any other features like which word is responsible for the particular sentiment class. Similarly, many images can be labelled and feed to the network to learn to identify if there are faces in the image or not. The network can achieve this without being explicitly giving face features like in the traditional algorithms

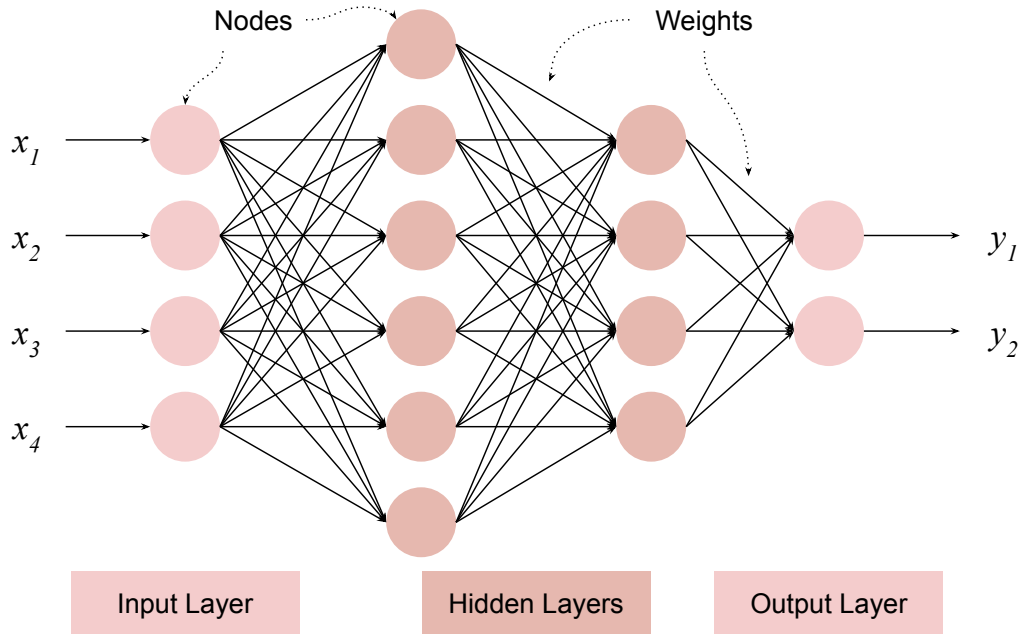


Figure 3.2: Example of Multi-Layer Perceptron Architecture.

(Lawrence et al., 1997). The traditional algorithms require facial features such as face landmark features, the shape of the eyes, nose, cheekbones, and jaw of the subjects' face. However, the neural network approach takes the images of the faces and learns to identify them from the given input image.

A Multi-Layer Perceptron (MLP) is a basic feedforward artificial neural network (ANN), which contains multiple layers of the perceptrons. It can be constructed with at least three layers: input layer, hidden layer and output layer; where all the nodes use a nonlinear activation function except for the nodes in the input layer. The nodes in all the layers are interconnected with weights. Figure 3.2 shows an example of MLP consisting an input layer (with four nodes), two hidden layers and an output layer (with two nodes). Commonly, MLP is trained with a supervised learning technique using backpropagation; learning to distinguish the data points that are not linearly separable. In several neural architectures, the feedforward neural network is mostly used as a bridge at the output layer, and hence we find it very useful in our experiments. At the output layer, a *sigmoid* or a *softmax* activation function is commonly employed with the respective cost function in numerous NLP tasks for binary or categorical classification respectively. We find the MLP layer beneficial with *softmax* function

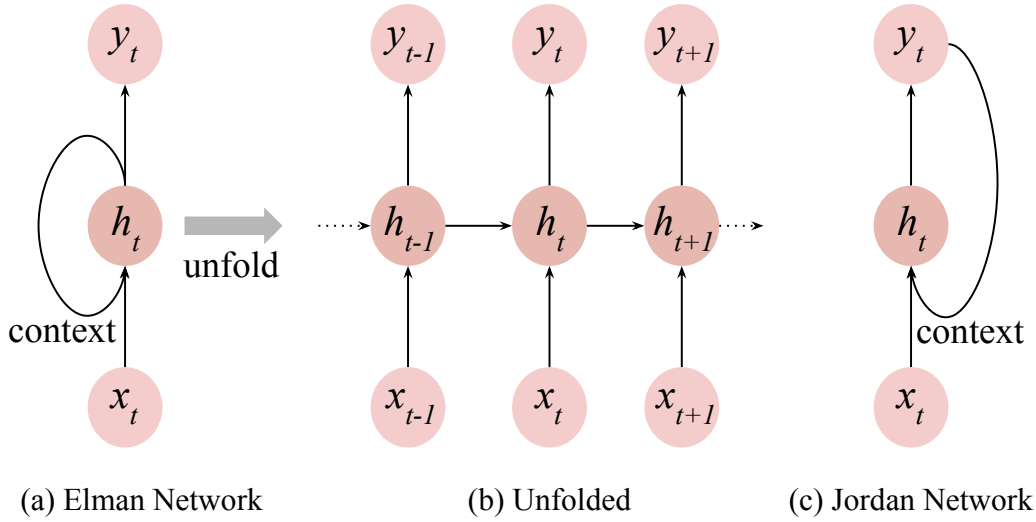


Figure 3.3: (a) Basic Elman RNN Architecture with its (b) Unfolded Structure and it is mostly compared with (c) Jordan Network.

for almost all the dialogue act and emotion recognition tasks in this thesis work.

3.2 Recurrent Neural Networks

Recurrent neural network (RNN) model contains a self-connected hidden layer providing an ability to pursue a memory of previous inputs (Elman, 1990; Wermter and Sun, 2000). The benefit of pursuing the memory of recent inputs in the network's hidden state allows it to learn from the past context in the sequential input points, as depicted in Figure 3.3. RNNs are most widely used to process sequential and time-series data. The idea of the RNN has emerged from connectionist modelling (Wermter, 1995). In natural language processing, context plays a crucial role in several stages. For example, looking at the characters, words, utterances, and conversations; they possess sequential feature representations. Hence making the RNN architectures most suitable for the NLP tasks, such as language modelling, text classification, summarising, and translation. We will explore different variants, mechanisms and types of the RNNs in the following sections.

3.2.1 Variants of RNNs

We present here two principal variants of the RNNs: Elman and Jordan networks.

The **Elman RNN** architecture is the fundamental and most widely used RNN variants, as shown in Figure 3.3(a) and with its unfolded structure in 3.3(b). Elman RNN uses hidden states as a context to further calculate the new hidden states (Elman, 1990). The hidden units (h_t) and output values (y_t) for the given input (x_t) are calculated with the following equations:

$$h_t = \sigma(W_h * x_t + U_h * h_{t-1} + b_h) \tag{3.1}$$

$$y_t = \sigma(W_y * h_t + b_y) \tag{3.2}$$

where W_h, U_h and W_y are weight respective matrices; b_h and b_y are respective hidden and output biases; and σ represents the *sigmoid* activation function.

The **Jordan RNN** architecture, on the other hand, uses the output as a context directly instead of the hidden states, as shown in Figure 3.3(c) (Jordan, 1997). The hidden units (h_t) and output values (y_t) for the given input (x_t) are calculated with the following equations:

$$h_t = \sigma(W_h * x_t + U_h * y_{t-1} + b_h) \tag{3.3}$$

$$y_t = \sigma(W_y * h_t + b_y) \tag{3.4}$$

where W_h, U_h and W_y are weight respective matrices; b_h and b_y are respective hidden and output biases; and σ represents the *sigmoid* activation function. Only a difference that the hidden state h_t is calculated using y_{t-1} instead of h_{t-1}.

3.2.2 LSTM and GRU Architectures

LSTM: One of the most successful and widely used architecture units of RNN cell is a Long Short-term Memory (LSTM) network (Hochreiter and Schmidhuber, 1997). LSTM is a distinctive architecture which is able to learn long-term dependencies with the help of its contextual feedback. It can process sequential and time-series data similar to the simple Elman RNN. LSTM is applicable to the tasks of contextual-predictive application with sparse data such as speech recognition (Li and Wu, 2015), scene and event detection (Fernando et al., 2018), handwriting recognition (Graves et al., 2008), and weather forecast.

The LSTM unit receives an encoded feature x_t as an input and outputs a hidden representation h_t, as shown in Figure 3.4 (Hochreiter and Schmidhuber, 1997). The hidden h_t and memory c_t vectors are derived from the input x_t and

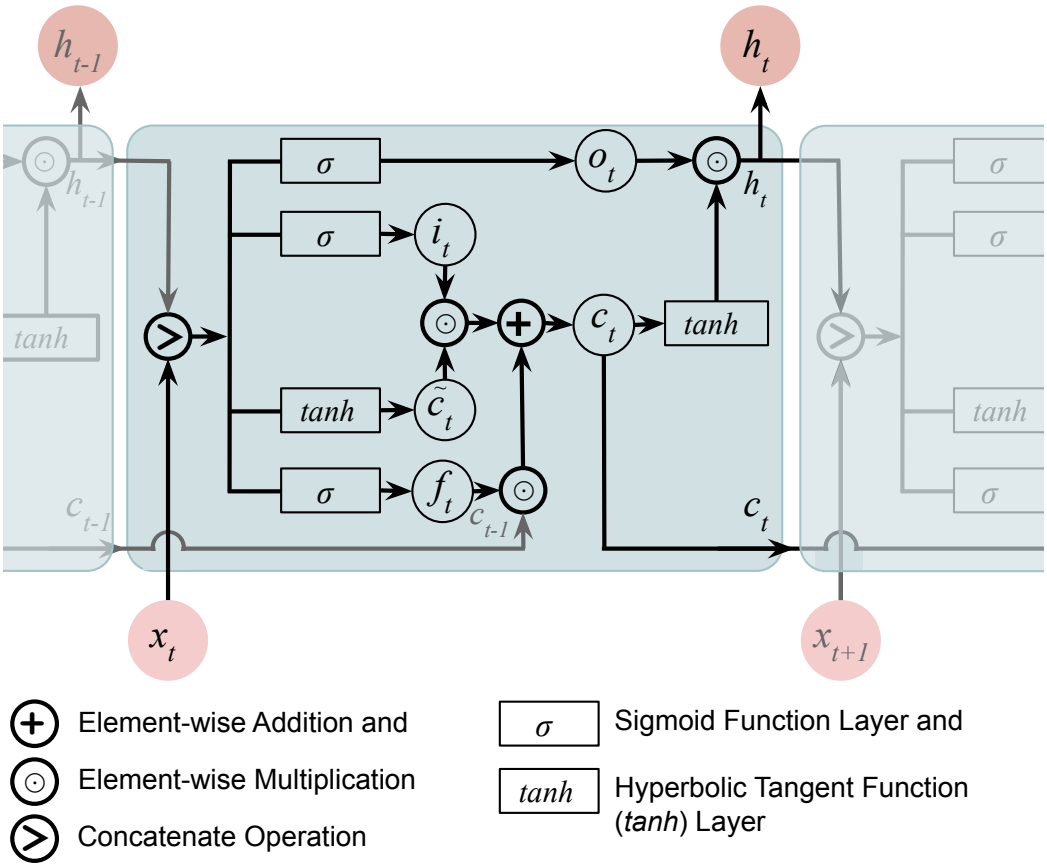


Figure 3.4: Long Short-term Memory (LSTM) Architecture.

past hidden vector h_{t-1} vectors, which are responsible to control state updates and outputs. The LSTM consists of a forget gate f, an input gate i, an output gate o, and a memory cell c, which are updated at time step t as follows:

$$f_t = \sigma\left(W_f * h_{t-1} + I_f * x_t + b_f\right) \tag{3.5}$$

$$i_t = \sigma\left(W_i * h_{t-1} + I_i * x_t + b_i\right) \tag{3.6}$$

$$o_t = \sigma\left(W_o * h_{t-1} + I_o * x_t + b_o\right) \tag{3.7}$$

$$\tilde{c}_t = tanh\left(W_c * h_{t-1} + I_c * x_t + b_c\right) \tag{3.8}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \tag{3.9}$$

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

where σ is the *sigmoid* function, W_f, W_i, W_o, W_c are the recurrent weight matrices, I_f, I_i, I_o, I_c are the corresponding projection matrices and b_f, b_i, b_o, b_c are

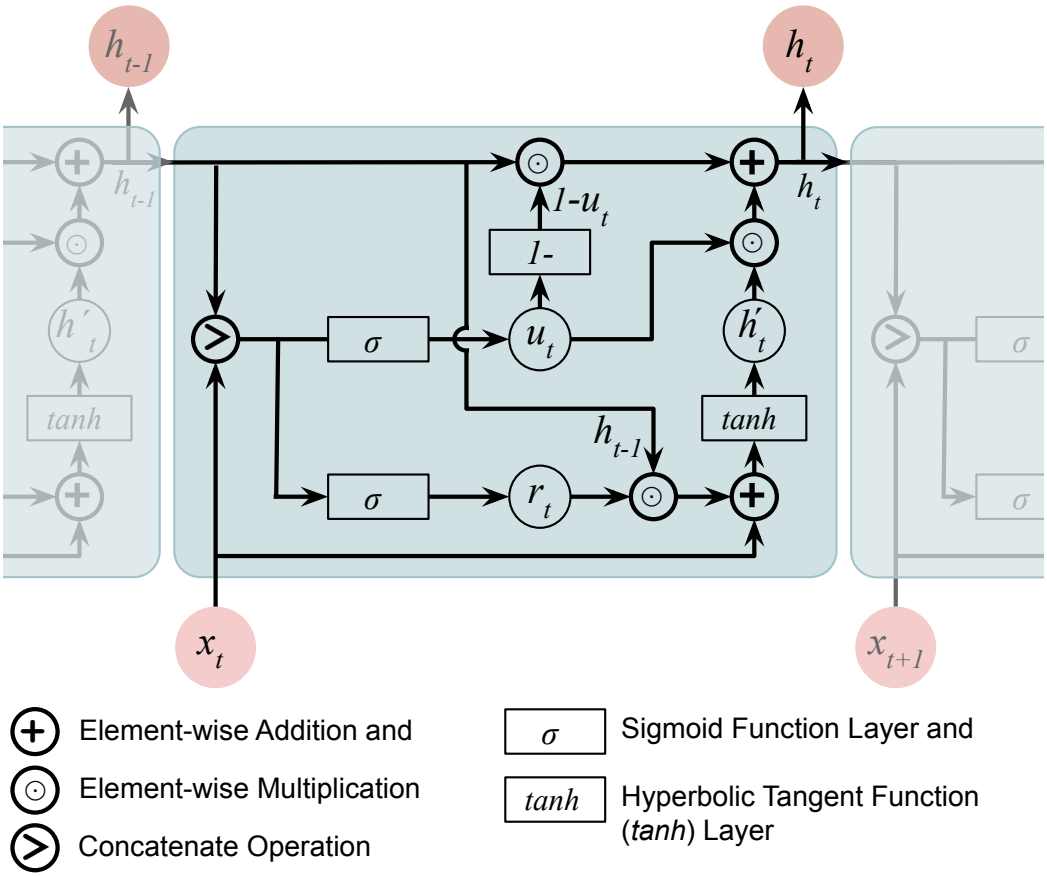


Figure 3.5: Gated Recurrent Unit (GRU) Architrcture.

respective learned biases. The weight-projection matrices and bias vectors are initialized randomly and learned during the training process. The gating functions of the LSTM helps this RNN architecture mitigate the vanishing and exploding gradient problems and train the model smoothly.

GRU: Another a very popular RNN unit architecture is Gated Recurrent Unit (GRU) network. Kyunghyun Cho et al. introduced GRU in 2014 (Cho et al., 2014) while proposing an RNN-based encoder-decoder architecture for the machine translation task. The GRU architecture unit is presented in Figure 3.5, and we can notice that it has fewer parameters than LSTM, as lacking the output gate; however, it performs similar to that of LSTM. GRU contains an update gate and a reset gate those work similar to LSTM gates. For example, the update gate decides on the information what to throw and add, similar to the forget and input gates in the LSTM cell. The reset gate acts similar to the forget gate, to

determine the past information to forget. The internal states can be computed using following equations:

$$u_t = \sigma\left(W_u * h_{t-1} + I_u * x_t + b_u\right) \tag{3.11}$$

$$r_t = \sigma\left(W_r * h_{t-1} + I_r * x_t + b_r\right) \tag{3.12}$$

$$h'_t = \sigma\left(W_h * h_{t-1} + I_h * (r_t \odot h_{t-1}) + b_h\right) \tag{3.13}$$

$$h_t = (1 - u_t) \odot h_{t-1} + u_t \odot tanh(h'_t) \tag{3.14}$$

where x_t is an input vector, and h_t being an output vector. h'_t: candidate activation vector which is subject to reset based on r_t a reset gate vector. Finally, the output vector h_t is calculated with the adjustment in the update gate vector u_t deciding how much to forget. σ is the *sigmoid* function, W_u, W_r, W_h are the recurrent weight matrices, I_u, I_r, I_h are the corresponding projection matrices, and b_u, b_r, b_h are respective learned biases.

3.2.3 Additional Mechanisms for RNNs

With the underlying RNN cell architectures explained above, there are additional forms which makes them suitable for different tasks and enhance their capabilities. The main two mechanisms that we use in our experiments are hierarchical and bidirectional RNNs.

Hierarchical RNNs: The stacked RNNs are used when there is information that can be perceived as hierarchical, for example, conversational dialogue text: where characters form words; words form utterances and utterances form conversations (El Hihi and Bengio, 1996). Stacking the RNNs helps to improve its capability to exploit long-range temporal dependency and finding the structural hierarchy in the data (Zhao et al., 2017). The hierarchical RNN is generally composed of stacked layers of the RNN units, and each layer contains several RNN cells. However, the final architecture varies according to specific applications. For example, the hidden to hidden layer units can be concatenated before further processing or used separately for different computation purposes. Figure 3.6(a) shows a two layer stacked HiRNN, where first layer produces hidden representations $h^1 = \{h^1_t, h^1_{t-1}, ...h^1_{t-n}\}$ and second layer produces $h^2 = \{h^2_t, h^2_{t-1}, ...h^2_{t-n}\}$. These representations further concatenated to produce a combined result (mostly

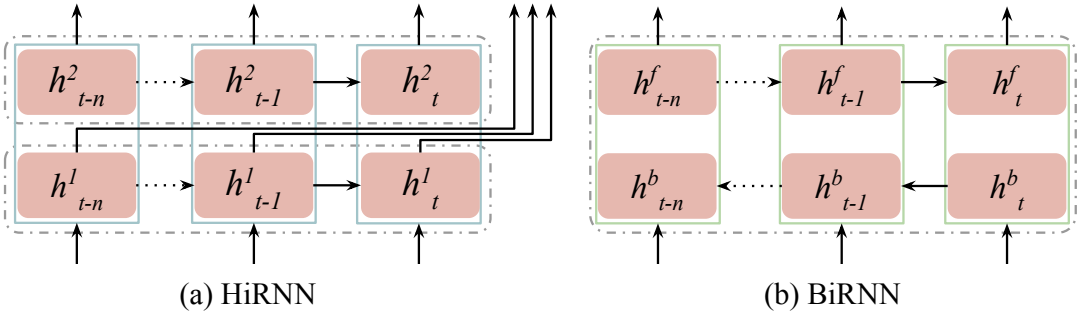


Figure 3.6: (a) Hierarchical RNN (HiRNN) Architecture, (b) Bidirectional RNN (BiRNN) Architecture.

in classification tasks) or can be used separately (generally feature extraction tasks). For example, in the first layer, h^1 hidden units can be used to capture word-level features, whereas h^2 hidden units can be used to capture utterance-level features.

Bidirectional RNNs: Another extended form of the unidirectional RNN is bidirectional RNN architecture. It introduces one extra hidden layer in the opposite direction (Schuster and Paliwal, 1997; Graves et al., 2013). That means the hidden to hidden layer connections flow into the opposite temporal direction. The model provides forward and backward states with corresponding directions of the hidden layers, as shown in Figure 3.6(b), and the final result is calculated as follows:

$$h_t^f = f\left(W_h^f h_{t-1}^f + W_u^f u_t + b_h^f\right) \tag{3.15}$$

$$h_t^b = f\left(W_h^b h_{t-1}^b + W_u^b u_t + b_h^b\right) \tag{3.16}$$

$$y_t = g\left(W_y^f h_t^f + W_y^b h_t^b + b_y\right) \tag{3.17}$$

where n is the number of inputs in the context for time instance t. W and h are the corresponding weight matrices and hidden vectors, where the superscripts f and b represent the forward and backward hidden layer directions respectively. In the unidirectional RNN model, there might be a chance that the model becomes more attentive to the current data point only, as sequential information is compressed to the final state. The BiRNN model, on the other hand, exploits the information in all given input data points by looking back and forth through them, looking at points in the input sequence uniformly. This ability makes the

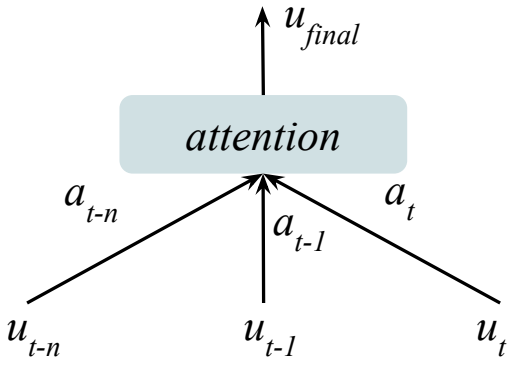


Figure 3.7: Attention Mechanism.

BiRNN model suitable for the conversational data. The utterances are treated uniformly to compute the utterance-level contributions using attention mechanism (more details in Section 4.5).

3.2.4 Attention Mechanism for RNNs

The attention mechanism is loosely based on the visual attention found in humans, and broadly used in image recognition and object tracking tasks (Larochelle and Hinton, 2010; Denil et al., 2012). Nevertheless, recently, attention mechanism with the RNNs are being used for several NLP tasks, such as machine comprehension and translation, and speech recognition (Bahdanau et al., 2015; Vinyals et al., 2015; Chorowski et al., 2015). In this thesis, we propose the attention mechanism to compute the contribution weights of the utterances for predicting the corresponding class as presented in Section 4.5. Given the number (n) of preceding utterances in an input sequence $u = \{u_t, u_{t-1}, ...u_{t-n}\}$. The attention layer computes the weights $a = \{a_t, a_{t-1}, ...a_{t-n}\}$ as the contribution for every corresponding input utterance in u, as depicted in Figure 3.7. Hence, the final utterance representation u_{final} formed with *attention* layer is calculated as:

$$m = tanh\left(W_h * u\right) \tag{3.18}$$

$$a = softmax\left(W_m^T * m\right) \tag{3.19}$$

$$u_{final} = tanh\left(u * a^T\right) \tag{3.20}$$

where W is a trained weight matrix while W^T being its transpose. We use the $softmax$ function to compute the weights which provides $\sum_{n=0}^{n} a_{t-n} = 1$. It is crucial for the utterance-level attention mechanism that we normalize a to

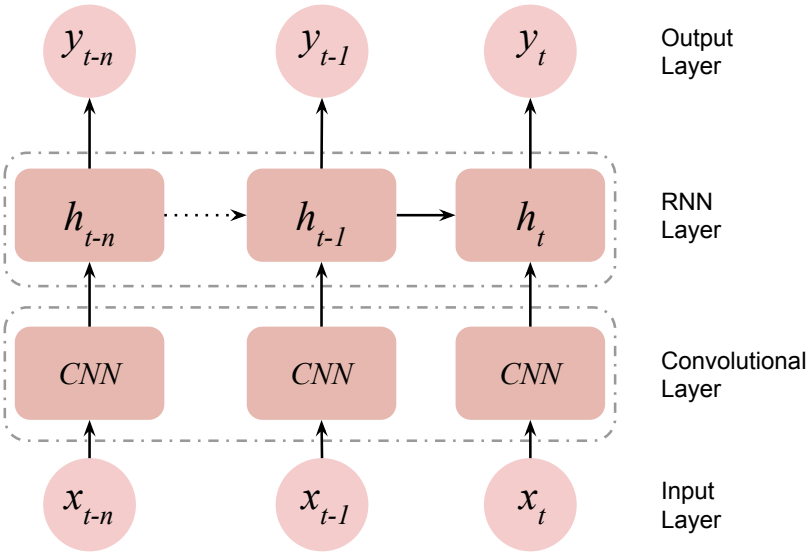


Figure 3.8: Recurrent Convolutional Neural Network.

interpret the amount of contribution for each utterance in u. This is just an example, a practical implementation will be followed in Section 4.5 for utterance-level attention mechanism.

3.2.5 Recurrent Convolutional Neural Networks

Convolutional neural networks (CNNs) have proven their capabilities in the computer vision applications (LeCun et al., 1998). However, recently they have been proven to the use cases in natural language processing (Kim, 2014; Zhang et al., 2015). Specifically, its combination with RNNs made them efficient and employable to the dialogue act recognition (Kalchbrenner and Blunsom, 2013b; Lee and Dernoncourt, 2016). Usually, CNNs are used to compress or encode the underlying features to high-level features.

For example, in NLP word embedding vectors of a sentence can be convoluted to form a final single vector representation which then can be used for further processing. As shown in Figure 3.8, consider the input x_t is an utterance in the conversation, consisting of word embedding vectors $x_t = \{w_1, w_2, ...w_n\}$ CNN convolutes over all these vectors and produces unique representations using kernels of a certain dimensionality, usually 1D, or 2D, that *slide* over the vectors to learn the underlying features. Convolution layer is then followed by *pooling* operation usually *max pooling*, not shown in the figure for simplicity, which takes

the convoluted feature map and chooses only maximum values. In this way, the CNN layer learns the relevant features and suppress irrelevant values to not pass on to the next layer.

The output features from the convolution layer are then used as input representations for the RNN layer. That takes advantage of receiving only CNN's relevant features, hence making RNNs learn faster. A practical example will be followed in Section 5.3, where we use such an architecture for contextual emotion recognition in dialogues.

3.3 Language Representations

NLP uses several techniques to convert the natural language into symbolic and numerical representations. A few of them we already mentioned previously: word embedding vectors and language models. These representations are useful when feeding them to the next neuronal layers such as CNNs or RNNs. There are also some other advantages of these features: they possess semantic and syntactic information.

3.3.1 Word Embeddings

Word embedding is one of the feature representation of language used to map the words in the form of real-valued vectors. In principle, it is a feature learning technique where words or phrases from the vocabulary of the given dataset are mapped to vector representations to learn meaningful aspects of the language. For example, Word2Vec is one of the most popular techniques to learn word vector representations for finding the similarity among the words (Mikolov et al., 2013a). Learning methods used in this approach are neural networks (Mikolov et al., 2013b), dimensionality reduction such PCA (Lebret and Collobert, 2013) and matrix factorization (Levy and Goldberg, 2014), and probabilistic models (Globerson et al., 2007).

The basic idea is to compute the embedding vector representation of the word with some constraints such as co-occurrence of words and learning to predict the word given neighbouring ones. Hence, no labelled data is required, and several relations can be learned using such an unsupervised approach; a few relations to mention, such as Male-Female, Verb tenses, and Country-Capital. The embedding

vectors are used to form a space with many dimensions, and potentially convert them to a continuous vector space with a much lower dimension. For example, the algorithms such as T-distributed Stochastic Neighbor Embedding (t-SNE) are generally used to convert such word embedding vectors into 2 or 3 dimensions to generate the visual representation (Maaten and Hinton, 2008).

Many of such word embedding vectors are available publicly; they are pre-trained on large corpora such as Word2Vec on Google News[1] or the Global Vectors (GloVe) (Pennington et al., 2014) on Wikipedia and Twitter[2]. The GloVe approach yields similar performance more efficiently using a co-occurrence matrix of the words in their context. On the other hand, FastText includes sub-word information to enrich word vectors and handle the out-of-vocabulary words (Bojanowski et al., 2017). Embeddings from Language Models (ELMo) is a particular case, as in this approach, language modelling technique is used to yield the embedding vector and claims to produce robust representations for out-of-vocabulary tokens using morphological clues (Peters et al., 2018). Bidirectional Encoder Representations from Transformers (BERT) prominently uses contextual learning approach to achieve similar results (Devlin et al., 2019). A few character-based embedding approaches, such as ELMo, unlike FastText, go even below the sub-word level. They use a context-based approach that is looking into the preceding and succeeding words aims.

One more form of the word embeddings is distributed as part of ConceptNet 5.5[3], which is created to represent the general knowledge in understanding the natural language. This embedding method allows the application to understand better the implications behind the words we use in general (Speer et al., 2017). It consists of a knowledge graph that relates the words and phrases with labelled edges. It helps in analyzing utterances when there is some new knowledge is required from the world. For example, humans are robust in processing even an ambiguous utterance as they integrate extrasentential knowledge, which results in improved comprehension (McCrae, 2010). While dealing with the conversational analysis, the word-level features are essential for analyzing the short utterances in a conversation. Hence, in our experiments, explained in Section 4.5, we use ConceptNet 5.5 word embedding representations over all the tokens in the ut-

[1] `https://code.google.com/archive/p/word2vec/`

[2] `https://nlp.stanford.edu/projects/glove/`

[3] `https://github.com/commonsense/conceptnet-numberbatch`

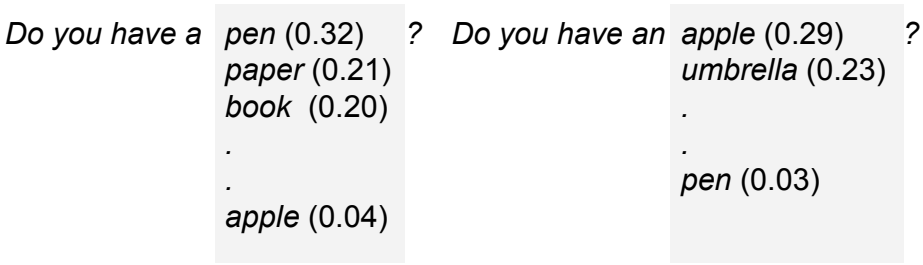


Figure 3.9: Example on Language Model.

terance. These embeddings also provide the out-of-vocabulary instance rate close to 10 per cent and mostly uncommon words for the Switchboard Dialogue Act Corpus.

3.3.2 Language Models: Character- and Word-level

When we deal only with words, we still compromise with some out-of-vocabulary instances, leading to the limitation of the feature representations. Especially for the short utterances in the conversation, it becomes crucial that we include all possible words, subwords, or even characters. This issue can be resolved using the character-level language modelling. First, we will define the language modelling technique. This technique was initially applied to the words (Jozefowicz et al., 2016). A language model determines a probability distribution P over a sequence of words, say of length n:

$$P(w_1, \ldots, w_n) \tag{3.21}$$

providing a contextual similarity notion to distinguish between words. For example, see the utterances (questions) in Figure 3.9, when a person says *"Do you have ..."*, the next characters *'a'* or *'an'* determines completely different probabilities. It is a data-driven model that learned from the data. For example, the article *'a'* is most likely to be used before words that start with a consonant sound and *'an'* before with a vowel sound. Hence, language models are able to compute probabilities of the appropriate next upcoming words. As a result, *"Do you have a..."* will most probably produce the list of words with higher probabilities that start with the consonant sound such as *pen*, *paper*, and *book*. Whereas, the utterance with *"an"* will produce the list of words with higher probabilities that start with the vowel sound such as *apple*, and *umbrella*.

The language models can also be trained at the character-level. We use one

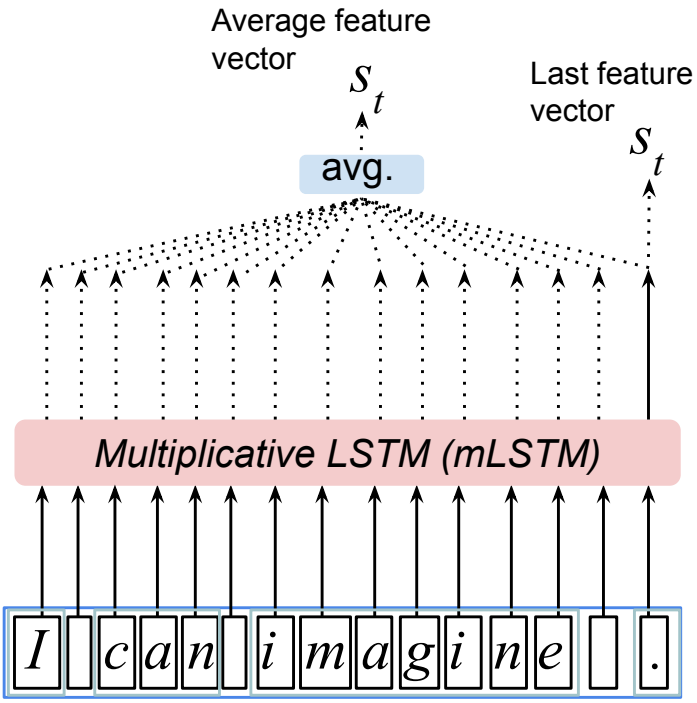


Figure 3.10: Character-level Language Model.

of such a character-level language model to encode the utterances in our experiments. It is a pre-trained character-level language model (LM), trained on ∼80 million Amazon product reviews[4] (Radford et al., 2017). The model consists of a single multiplicative long-short-term memory (mLSTM) network (Krause et al., 2016) layer to predict the next character given a set of the preceding characters, as shown in Figure 3.10. It was trained to generate reviews and authors discovered a sentiment neuron in the learned model. It takes the input characters sequentially to the mLSTM layer, and the hidden vector is obtained after the last character. An average vector can be also be produced using all hidden units of the mLSTM layer per character. We use these feature vector representations to encode the utterances in the experiments of dialogue act recognition (Bothe et al., 2018b,d), and emotion detection (Lakomkin et al., 2017).

[4] https://github.com/openai/generating-reviews-discovering-sentiment

3.4 Summary

In this chapter, we reviewed the fundamental neural techniques that are used in this thesis. Recurrent neural networks are considered as a backbone of natural language processing. We presented RNNs with its variant architectures such as Elman and Jordan networks. We also detailed the particular forms of the RNNs: LSTMs and GRUs, the most widely used in the NLP tasks. Then we explored various mechanisms of the RNNs: hierarchical and bidirectional RNNs. We also show how the attention mechanism and convolutional neural network can be used with the RNNs. Finally, we looked into the language representations: word embeddings and language models. Especially, the character-level language model where one of the RNN architecture is used to train the model, and we deploy it in our experiments to encode the utterances.

Chapter 4

Contextual Dialogue Act Recognition using RNNs

Dialogue act recognition is a necessary process of natural language understanding for any dialogue system and also often used for discourse and conversational analysis. Unlike emotion and sentiment plays a vital role in the decision-making process, the dialogue acts represent the meaning of the utterances, as explained in Section 2.4 "Natural Language Understanding (NLU) for HRI". The linguistic feature, like the dialogue act, can be used in a dialogue system for taking situation-aware actions based on different functions and intentions found in the utterance. For example, commonly, if the utterance dialogue act is a *question*, the conversational action could be an *answer*, or if it is a *proposal* then the action could be *acceptance* or *rejection*.

4.1 Introduction

The capability to analyze discourse compositionality is a crucial step towards understanding the conversational dialogue. The dialogue act (DA) recognition is the first level of investigation approaching conversational analysis (Stolcke et al., 2000). In most cases, DA recognition is considered a lexical-based or syntax-based classification at an utterance level. However, the discourse compositionality is a context-sensitive process, that means the DA of an utterance can be elicited from the preceding utterances (Grosz, 1982). Hence, classifying dialogue acts only at the utterance-level is not sufficient because their DA class arises from the context of the preceding utterances. For example, the utterance containing only the word

'yeah' might appear in several DA classes such as *Backchannel*, *Yes-answer*, and *Agree/Accept*, see the examples in Table 4.1. For such DA classes, the utterances are short, and mostly share similar lexical and syntactic cues (Jurafsky et al., 1998).

In this chapter, we explore RNN techniques to recognize the dialogue acts of the utterances in a contextual manner and apply them for different analytical scenarios. In our experiments, we use spontaneous spoken utterances from the Switchboard Dialogue Act (SwDA[1]) corpus. Please visit the complete list of the dialogue act labels with their tags and statistics in Appendix Table B.1. We first investigate the process of how dialogue act corpora are annotated, and the learning approaches used so far, detailed investigation in Section 4.2. We find that the dialogue act is context-sensitive within the conversation for most of the dialogue act classes. Nevertheless, previous models of dialogue act recognition work are on the utterance-level, and only very few consider the context. First, we show an utterance-level classification of the DAs using character-level language model and word-embedding utterance representations in Section 4.3. We propose a novel context-based learning method to classify dialogue acts using recurrent neural networks (RNNs), and we notice a significant improvement, see Section 4.4. We use the RNN architecture, Elman network, as explained in Section 3.2.1, for context learning of the discourse compositionality.

The results show that consideration of the preceding utterances as a context of the current utterance improves the dialogue act recognition accuracy. We found out that using at least three number utterances in the context produces better overall accuracy on the SwDA test set (Bothe et al., 2018d), provided in Section 4.4.1. We analyze the hidden internal states of the RNNs and plot them on 2D, and found interesting cluster formations. For example, we could see the clusters of the DA classes on the plotted graph, in Section 4.4.2. With this method we rank third globally for Dialogue Act Classification on Switchboard corpus according online records for state-of-the-art results[2], presented in Table 4.2. However, the first two (Kumar et al., 2018; Chen et al., 2018b) uses future and past utterances in the context, whereas we use only a few (three) past utterances, and achieve indifferent accuracy. It is essential to notice that we cannot listen to a future utterance in real-world before it is uttered unless one is looking at others'

[1]Available at `https://github.com/cgpotts/swda`

[2]`https://nlpprogress.com` and `https://paperswithcode.com`

conversation. Moreover, technically, the operational function of the dialogue system can not have access to future utterance in advance. In other words, once a response is generated by the dialogue system, it would be indifferent to consider correcting after the second turn has been finished. On the contrary, it can be used to improve the next turn response in case of a correction. Hence, we give much importance to using only a few of the preceding utterances in the context.

Conversational discourse analysis is another crucial task for natural language understanding and building a natural spoken dialogue system. The speech acts are mostly used to perform discourse analysis of the conversation, which are context-sensitive, where the context provides information for appropriate interpretation

Speaker	**Dialogue Act**	**Utterance**
A	Backchannel	Uh-huh.
B	Statement	About twelve foot in diameter
B	Abandoned	and, there is a lot of pressure to get that much weight up in the air.
A	Backchannel	Oh, yeah.
B	Abandoned	So it's interesting, though.
		...
B	Statement-opinion	it's a very complex, uh, situation to go into space.
A	Agree/Accept	Oh, yeah,
		...
A	Yes-No Question	You never think about that do you?
B	Yes-answer	Yeah.
A	Statement-opinion	I would think it would be harder to get up than it would be
B	Backchannel	Yeah.

Table 4.1: Example of a labeled conversation (portions) from the Switchboard Dialogue Act (SwDA) corpus.

(Sbisà, 2002). However, once the context is taken into account, a few questions related to the context could be answered. For example, how many utterances in the context contribute to the current utterance? How do context-utterances affect the interpretation (Austin, 1962; Searle, 1979; Wermter and Löchel, 1996; Sbisà, 2002)? We propose a context model based on an utterance-level attention mechanism using bidirectional recurrent neural network (Utt-Att-BiRNN) architecture to analyze the importance of preceding utterances while classifying DA for the current utterance, given in Section 4.5. In our setup, the BiRNN model is provided with the set of current and preceding utterances as input (Graves et al., 2013; Schuster and Paliwal, 1997; Bahdanau et al., 2015; Zhou et al., 2016).

Our model outperforms previous models that use only preceding utterances as context on the used corpus. This model is intended to not only create the context-based learning architecture but also to analyze the amount of contributing information in the utterances for the dialogue act recognition task. As a result, we investigated the discourse analysis in a conversation with context-based

Models	**Accuracy**	**Context method**
CRF-ASN (Chen et al., 2018b)	81.30%	Former and later utterances in context with data-dependant utterance representations
Bi-LSTM-CRF (Kumar et al., 2018)	79.20%	Preceding utterances and dialogue acts, model makes prediction of all DAs for the given set of utterances using word embedding representations
RNN with LM (Bothe et al., 2018d)	77.34%	Only preceding utterances in context with data-independent language model utt-representation

Table 4.2: Results compared with the state of the art from nlpprogress.com on the Switchboard Dialogue Act corpus, available at `https://nlpprogress.com/english/dialogue.html`, at the time of writing the article Bothe et al. (2018d).

learning using the proposed model. We discover that many instances of the DAs are detected wrongly with both models. These instances are reported along with the sample examples where the utterance-level model fails to predict correctly against the Utt-Att-BiRNN model. The examples from SwDA corpus test set are reported where both the models fail to detect the DAs, given in Section 4.5.2. We can also determine ambiguously or wrongly annotated utterances with this investigation. In the given dataset, we show that context-based learning not only improves the performance but also achieves the higher confidence toward recognition of dialogue acts, given in Section 4.5.3. As said above, another contribution of these experiments is a mechanism to discover the amount of information each utterance contributes to classify the subsequent one, which answers one of the above questions. We found that when classifying short utterances, the closest preceding utterances contribute to a higher degree, provided in Section 4.5.4.

Hence in this chapter, we explore a detailed insight into the annotation and modelling of the dialogue acts. In the first scenario, we propose a neural model for discourse analysis within the context of a conversation using RNNs, modelled only with preceding utterances. This model uses utterances represented by the character-level language model and word-embeddings trained on domain-independent data. As stated earlier, we evaluated the proposed models on the Switchboard Dialogue Act (SwDA) corpus and showed how using context affects the results. We show that the context of using only a few preceding utterances makes the model suitable for a real-time dialogue system, in contrast to the models where the whole conversation is used as an input. We perform qualitative and quantitative analyzes on the conversational data.

In the second part, we present an attention-based bidirectional RNN model to perform further conversational analysis that helps to answer a few key questions. How many preceding utterances in the context are required to recognize the DA of current utterance sufficiently? How much each preceding utterance in the context contributes while recognizing the DA of the current one? We also show that context-based models detect DAs correctly with higher confidence than the utterance-level model (especially in the cases where context plays an important role). Such analysis shows the effectiveness of the utterance-level context-model over without context model.

4.2 Annotation and Modelling Background

In this section, we will discuss the first issue from the above discussion, which provides a detailed insight into the annotation and modelling of the dialogue acts, and hence we will be looking into the background of the related work.

4.2.1 Annotation of Dialogue Act (DA) Corpora

Annotation Process and Standards: Research on the dialogue acts became prominent with the commercial reality of spoken dialogue systems. There have been many taxonomies: speech act (Austin, 1962), which was later modified into five classes (Assertive, Directive, Commissive, Expressive, Declarative) (Searle, 1979). One of the most widely used taxonomy is the Dialogue Act Markup in Several Layers (DAMSL) tag set. It has labels divided into four primary categories: (1) Communicative-Status, (2) Information-Level, (3) Forward Looking Function and (4) Backward Looking Function (Allen and Core, 1997). Communicative-Status contains the dialogue act labels that provide flags for the utterance intelligible and successful completion. Information-Level dialogue acts provide tags for informing what is happening in the utterance semantically, for example, a task being carried out or communication indications like greetings and acknowledgements. Forward Looking Function includes the DA labels for Statement (with and without opinion), Info-request, Thanking whereas Backward Looking Function such as Accept, Reject, and Answers. Many such standard taxonomies and schemes exist to annotate conversational data, and most of them follow the concept of discourse compositionality. The DAMSL scheme is one of the essential tag sets for analyzing dialogues and building a natural dialogue system (Skantze, 2007).

Corpus Insight: We have investigated the annotation method for two corpora: Switchboard (SWBD) (Godfrey et al., 1992; Jurafsky et al., 1997) and ICSI Meeting Recorder Dialogue Act (MRDA) (Shriberg et al., 2004). Both of these datasets are annotated with the DAMSL tag set. The annotation includes not only the utterance-level but also segmented-utterance-level labels. For example, the utterances of the speaker B in the first portion in Table 4.1 are segmented and annotated for two different dialogue act labels, Statement and Abandoned. MRDA corpus contains multiple conversation partners in the meeting scenarios, whereas we target two-speaker conversation; hence we use the Switchboard dialogue act

	Training Samples	Testing Samples
Number of Conversations	1,115	19
Number of Utterances	196,258	4,186

Table 4.3: Switchboard Dialogue Act corpus details.

corpus for the experiments. The DAMSL tag set provides very fine-grained and detailed DA classes and follows the discourse compositionality. For example, the SWBD-DAMSL is the variant of DAMSL specific to the Switchboard corpus. The Switchboard DAMSL Coders Manual[3] can be followed for knowing more about the dialogue act labels, and the detailed statistic is given in Appendix B.1. It distinguishes WH-questions (*qw*), Yes-No Questions (*qy*, Open-ended (*qo*), and Or-questions (*qr*) classes, not just because these questions are syntactically distinct, but also because they have different backward functions (Jurafsky, 1997). The *qy* dialogue act labeled utterance is more likely to get a *"yes"* or *"no"* answer than a *qw*. It also gives an intuition that the answers follow the syntactic formulation of the question, which provides a context. For example, *qy* is used for a question that from a discourse perspective expects either Yes-answer (*ny*) or No-answer *nn* dialogue act class. The statistics of the Switchboard Dialogue Act corpus is given in Table 4.3.

Nature of Discourse in Conversation: The dialogue act is a context-based discourse concept that means the DA class of a current utterance can be derived from its preceding utterance. We will elaborate on this argument with an example given in the last portion of the conversation in Table 4.1. Speaker *B* utters *"Yeah."* twice in the given portion of dialogue, and each time it is labelled with two different DA labels. It is simply due to the context of the previously conversed utterance. If we see the four utterances of the example, when speaker *A* utters the *Yes-No Question* DA class, speaker *B* answers with *'yeah'* which is labelled as *Yes-answer* DA class. However, after the utterance with *Statement-opinion* DA class, the same utterance *"yeah"* is labelled as *Backchannel* and not *Yes-answer*. It provides strong evidence that when we process the text of a conversation, we can see the context of a current utterance in the preceding utterances only, and we never watch the future utterances during the annotation process.

[3] `https://web.stanford.edu/~jurafsky/ws97/manual.august1.html`

Prosodic Cues for DA Recognition: It has also been noted that prosodic knowledge plays a significant role in DA identification for certain DA types (Jurafsky et al., 1998; Stolcke et al., 2000). The main reason is that the acoustic signal of the same utterance can be very different for different DA classes. It indicates that if one wants to classify DA classes only from the text, the context must be an indispensable aspect to consider: simply classifying single utterances might not be enough, but considering the preceding utterances as a context is essential. However, the acoustic signals can be used independently or ensembled to reasonably recognize the utterances' DA classes.

4.2.2 Modelling Approaches

Lexical, Prosodic, and Syntactic Cues: Many studies have been carried out to find out the lexical, prosodic and syntactic cues for the DA recognition task (Stolcke et al., 2000; Surendran and Levow, 2006; O'Shea et al., 2012; Yang et al., 2014). For the SwDA corpus, the state-of-the-art baseline result was 71% for more than a decade using a standard Hidden Markov Model (HMM) with language features such as words and n-grams (Stolcke et al., 2000). The inter-annotator agreement accuracy for this corpus is 84%, and in this particular case, we are still far from achieving human accuracy. However, as we saw the words like *"yeah"* appear in the utterances of several DA classes such as *Backchannel*, *Yes-answer*, *Agree/Accept* etc. Hence, the prosodic cues play a crucial role in identifying the DA classes, as the same utterance can acoustically differ a lot which helps to distinguish the specific DA classes (Shriberg et al., 1998). Several approaches, like traditional Naive Bayes and HMM models, use minimal information and ignore the dependency of the context within the conversation (Grau et al., 2004; Tavafi et al., 2013). They achieved 66% and 74.32% respectively on the SwDA corpus test set.

Utterance-level Classification: Perhaps most research in modelling dialogue act identification is conducted at utterance-level (Stolcke et al., 2000; Grau et al., 2004; Tavafi et al., 2013; Ji et al., 2016; Lee and Dernoncourt, 2016). The emerging advances in deep learning also yielded a significant impact on the DA recognition task. In natural language conversation, most utterances are very short; hence it is also referred to as short text classification problem.

A Novel Approach - Context-based Learning: Classifying the DA classes

at single utterance-level might fail when it comes to DA classes where the utterances share similar lexical and syntactic cues (words and phrases) like the *Backchannel*, *Yes-answer* and *Accept/Agree* classes. Some researchers proposed the utterance-dependent context-based learning approaches (Kalchbrenner and Blunsom, 2013b; Bothe et al., 2018d,b; Kumar et al., 2018; Liu et al., 2017; Ji et al., 2016; Tran et al., 2017; Ortega and Vu, 2017; Meng et al., 2017). The context-based learning approach was first proposed to model discourse within a conversation using RNNs (Kalchbrenner and Blunsom, 2013b). The DA of the current utterance was computed using the preceding utterances as a context, achieving state-of-the-art results of about 74% accuracy on the SwDA corpus test set (Kalchbrenner and Blunsom, 2013b; Ortega and Vu, 2017). Kalchbrenner and Blunsom (2013b); Ortega and Vu (2017) have proposed context-based learning, where they represent the utterance as a compressed vector of the word embeddings using CNNs and use these convoluted utterance representations to model discourse within a conversation using RNNs. Their architecture also gives importance to turn-taking by providing the speaker identity but does not analyze their model in this regard. This approach achieves about 73.9% accuracy on the SwDA corpus test set. Lee and Dernoncourt (2016) also use recent techniques such as RNNs and CNNs with word-level feature embeddings and achieve about 73% of accuracy.

In other approaches, a hierarchical convolutional and recurrent neural encoder model is used to learn utterance representation by processing the whole conversation (Kumar et al., 2018; Liu et al., 2017). The utterance representations are further used to classify DA classes using the conditional random field (CRF) as a linear classifier. The model can scan the past and future utterances at the same time within a conversation, which limits its usage in a real-time dialogue system where the system can only perceive the preceding utterance as a context but does not know the upcoming utterances. In this research line, the context-based learning approach processes the whole set of utterances in a conversation, where the model can see past and future utterances to calculate the DA of the current utterance (Ji et al., 2016; Kumar et al., 2018). Ji et al. (2016) use discourse annotation for the word-level language modelling on the SwDA corpus and achieve about 77% of accuracy but also highlight a limitation that this approach is not scalable to large data. On the other hand, this work suggests that a domain-independent language model which is trained on the big data might

be a solution. When building a dialogue system, for example, in human-machine interaction scenario, one can only perceive the preceding utterance as a context but does not know the upcoming utterances. The DA corpus is also annotated by looking at the preceding utterances (Godfrey et al., 1992). Therefore, we use a context-based learning approach where only preceding utterances are considered and regard the 73.9% accuracy (Kalchbrenner and Blunsom, 2013b; Ortega and Vu, 2017) on the SwDA corpus as a state-of-the-art result for this particular task.

4.3 Utterance Representation and No-context DA Recognition

Before proceeding to the context-based learning, we will see the choices of the utterance representation and baseline model to classify the DAs at utterance-level. Then we can decide which representation to use for the dialogue act recognition and conversational analysis using the neural models. A certain number of words constitute the utterance, and certain characters represent the word. We have to encode the utterances either on the word-level or character-level feature representations. Character-level encoding allows processing words and whole sentences based on their smallest units and still capturing punctuation and permutation of words (Jozefowicz et al., 2016). Word-level encoding allows us to capture semantic-level information and go beyond the character-level units. On the other hand, the character-level encoding can easily achieve vector representation for an out-of-vocabulary word. Whereas, word-embedding representation can only fetch the vector representation of the seen words. Vector representation of the out-of-vocabulary words from character-level encoding can be learned when further fed to the specific task-learning model. However, when the word-level vector representation is just not available (in case of the out-of-vocabulary word in the word-embedding dictionary), the model has to keep track of these new words. These words representations have to be learned on the fly and stored separately, creating another dictionary of word vectors. The following section explains the used utterance representations in our experiments and the utterance-level DA recognition model.

Character-level Representations: The character-level utterance is encoded

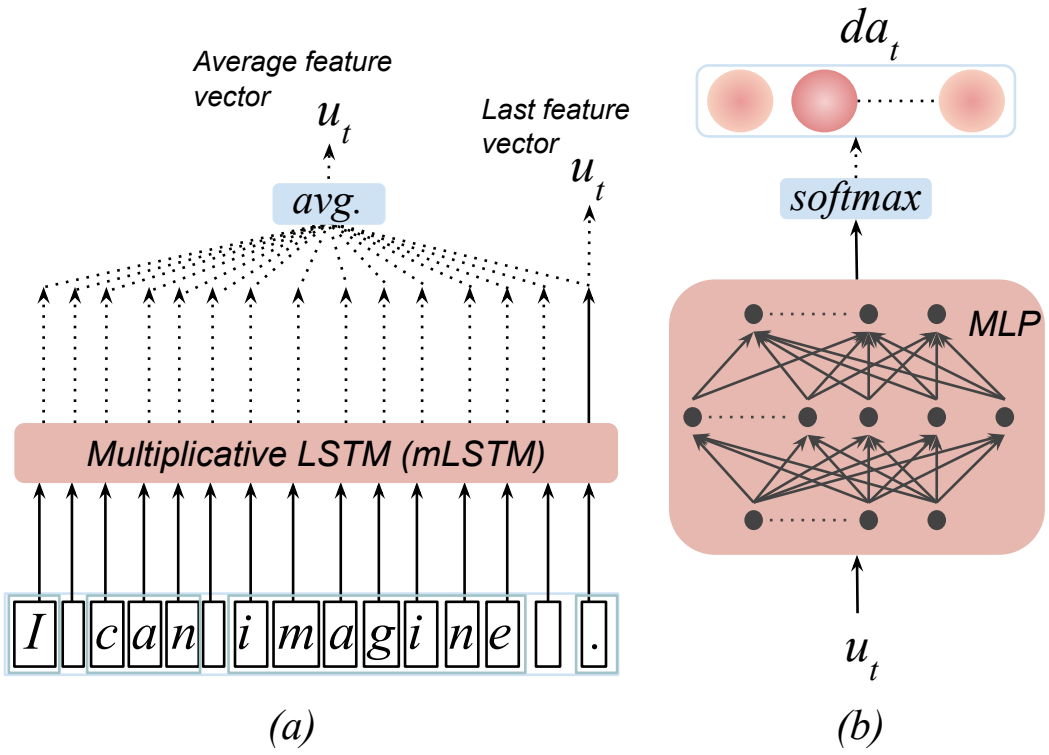


Figure 4.1: (a) Multiplicative LSTM (mLSTM) character-level language model to produce the sentence representation u_t. The character-level language model is pre-trained and produces the feature (hidden unit states of mLSTM at the last character) or average (average of all hidden unit states of every character) vector representation of the given utterance. (b) Utterance-level classification using MLP layers with a *softmax* function (our baseline model).

with a pre-trained character-level language model (character-LM)[4] (Radford et al., 2017) as shown in Figure 4.1(a). This model consists of a single multiplicative long-short-term memory (mLSTM) network (Krause et al., 2016) layer with 4,096 hidden units. The mLSTM is composed of the LSTM and multiplicative RNN units, and it considers each possible input in a recurrent transition function and trained as a character-level language model on ∼80 million Amazon product reviews (Radford et al., 2017). We sequentially input the characters of an utterance to the mLSTM and get the hidden vector obtained after the last character and average the states' overall characters in the utterance.

Utterance-level DA Classification with Character-LM: We use the

[4] `https://github.com/openai/generating-reviews-discovering-sentiment`

Model input	Accuracy
Most common class	31.50%
Stolcke et al. (2000)	71.00%
Last feature vector	71.48%
Average feature vector	**73.96%**
Concatenated vector	73.18%

Table 4.4: Accuracy of the dialogue act recognition using the character-LM utterance representation for 42 dialogue act classes.

character-LM utterance representation encoded with the pre-trained character language model explained above. This model consists of a single mLSTM network layer with providing a vector of size 4,096. All the utterances from SwDA corpus are encoded with this method. The hidden vector representations obtained after the last character (Last feature vector) and the average vector representations for all the characters (Average feature vector) are extracted and used for the training and testing purpose. We also concatenate these two representations to create another vector representation (Lakomkin et al., 2017).

We classify these representations with a feedforward neural network (FNN) consisting of MLP layer (discussed in Chapter 3), as shown in Figure 4.1(b). We use only one layer with 64 hidden units in this FNN-model. Finally, the $softmax$ function is used to compute probability distribution of the dialogue acts (da_t) for the input utterance (u_t). The results are given in Table 4.4, and we can see that the average vector seems to carry more information related to the DA; hence we use it for future experiments. The concatenated vector representation does not seem to improve the results any further. On the other hand, it shows an advantage of using domain-independent data: it is rich regarding features being trained on big data, perhaps surpassing scalability limitation as mentioned in Ji et al. (Ji et al., 2016). Hence we use these domain-independent character-LM representations for our proposed context-based learning approach.

Word-level Representations: Word-level features are essential for analyzing the short sentences of utterances in a conversation. We have various word-embedding distributions to use in our experiments such as ConceptNet, word2vec,

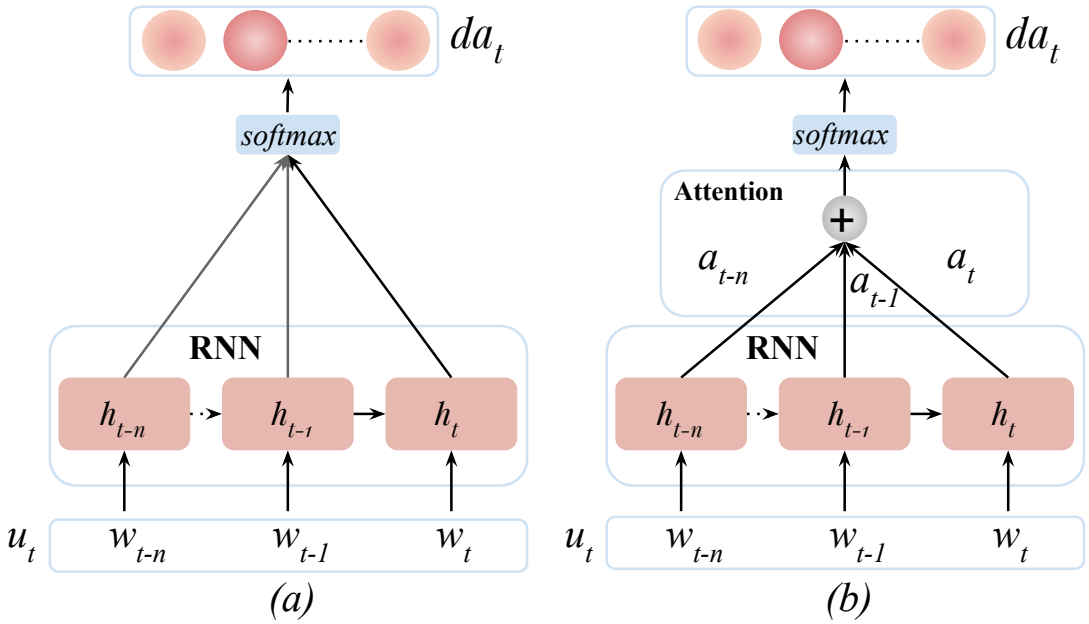


Figure 4.2: The utterance-level RNN setup for learning the dialogue act recognition with word embeddings. (a) Word-embedings ($w_t ... w_{t-n}$) of utterance u_t fed directly to RNN architecture, and (b) same with attention mechanism, to classify DAs (da).

GloVe, and ELMo. We chose ConceptNet and ELMo as they are among the language models trained on the natural language text corpus.

ConceptNet 5.5[5], as discussed in Section 3.3.1, is designed to represent the general knowledge in understanding the natural language and allows the application to understand better the implications behind the words people use (Speer et al., 2017). The embedding dimension of ConceptNet used in this experiment is 300 and averaged over all tokens in the utterance. These embeddings provide the out-of-vocabulary instance rate close to 10 per cent and mostly for uncommon words. ELMo (Embeddings from Language Models)[6] models complex characteristics of words and their variation across linguistic contexts also knows as polysemy. The word embedding representations are learned internal states of bidirectional language model (Peters et al., 2018). Each word in the utterance is represented with an embedding vector of dimension 1024 from a pre-trained language model

[5] https://github.com/commonsense/conceptnet-numberbatch
[6] https://allennlp.org/elmo

Model input	Accuracy
Most common class	31.50%
Stolcke et al. (2000)	71.00%
FNN-model (Mean rep. ConceptNet)	71.73%
FNN-model (Mean rep. ELMo)	72.59%
FNN-model (Concatenated rep.)	70.83%
RNN-model (ConceptNet)	72.32%
RNN-att-model (ConceptNet)	72.29%
RNN-model (ELMo)	74.92%
RNN-att-model (ELMo)	**75.13%**

Table 4.5: Accuracy of the dialogue act recognition using the word embedding utterance representations for 42 dialogue act classes.

on a large corpus. These embeddings provide a minimal number of the out-of-vocabulary instance.

Utterance-level DA Classification with Word-level Representations: In the case of word embedding representations, we have two ways to perform the utterance-level classification of the DA classes. First, we take the mean of the word vectors over the utterances and feed those averaged vectors to FNN-model, as shown in Figure 4.1(b). We average ConceptNet and ELMo embeddings and found that ELMo embeddings show better performance over ConceptNet, as can be seen in Table 4.5.

Second, we use RNNs to model the recurrency in the word sequence and classify the DA classes, as shown in Figure 4.2(a). In this case, we can also adapt to using attention mechanism on top of the RNN model, as shown in Figure 4.2(b). In this way, we acheive a mechanism to compute the contribution $(a_t, a_{t-1}, ..., a_{t-n})$ of all words $(w_t, w_{t-1}, ..., w_{t-n})$ in the utterance (u_t) towards detecting a particular dialogue act (da_t) using the $softmax$ function. The results of these models are presented in Table 4.5, we can see that ELMo embeddings again outperform the ConcpetNet embedding representations. The attention mechanism does not seem to add any performance to the accuracy but provides a way to compute where

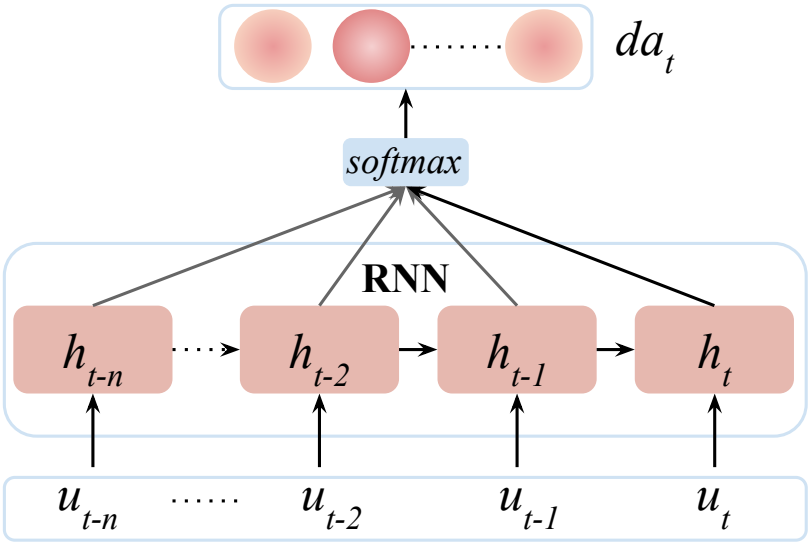


Figure 4.3: The RNN setup for learning the dialogue act recognition with the previous sentences as context. u_t is an utterance representation derived with a character-level language model and has a dialogue act label da_t. u_{t-1} and u_{t-2} are the preceding utterances of u_t. The RNN is trained to learn the recurrency through previous utterances u_{t-1} and u_{t-2} derived as h_{t-1} and h_{t-2} as a context to recognize the dialogue act of current utterance u_t which is represented by h_t used to detect da_t.

the network is extra attentive on the input features (Kumar et al., 2018). Hence, we use ELMo embeddings in the conversational analysis part of this chapter, in Section 4.5. However, in the next experiment, we will use only character-LM for showing how many preceding utterances are useful for computing the best overall accuracy of the SwDA corpus test set.

4.4 Context Learning of DAs using RNNs

In this experiment, we will use only character-LM utterance representation to simplify the problem statement towards the solution. The context-based learning is applied with the help of RNNs. As shown in Figure 4.3, the utterances with their character-level language model representation u_t are fed to the RNN with the preceding utterances (u_{t-1}, u_{t-2},...,u_{t-n}) being the context. Hence, n represents the number of utterances in the context. In this case RNN gets the input u_t, and

stores the hidden vector h_t at time t (instance of utterance in the conversation) (Elman, 1990), which is calculated as:

$$h_t = f\left(W_h * h_{t-1} + I * u_t + b\right) \tag{4.1}$$

where $f()$ is a *sigmoid* function, W_h and I are the recurrent and input weight matrices respectively and b is a bias vector learned during training. h_t is computed using the previous hidden vector h_{t-1} which is computed in a same way for preceding utterance u_{t-1}. The output da_t is the dialogue act label of the current utterance u_t calculated using h_t, as:

$$da_t = g\left(W_{out} * h_t\right) \tag{4.2}$$

where W_{out} is the output weight matrix. The weight matrices are learned using back-propagation through time. The task is to classify several classes; hence we use the $softmax$ function $g()$ at the output layer. The input is the sequence of the current and preceding utterances, for example, u_t, u_{t-1}, and u_{t-2}. We reset the RNN when it sees the current utterance u_t. We also provide the speaker identification information to the network to find the change in the speaker's turn in the conversation. The speaker id 'A' is represented by [1,0] and id 'B' by [0,1] and it is concatenated with the corresponding utterances u_t, shown in Figure 4.4.

The Adam optimizer (Kingma and Ba, 2014) was used during training the network with a learning rate of $1e-4$, which decays to zero as training progresses, and clipping gradients at norm 1. Early stopping was used to avoid over-fitting the network, with 15% of the training samples for validation. In all the learning cases, we minimize the categorical cross-entropy.

4.4.1 Results with RNNs - Number of Context Utterances

We follow the same data split of 1115 training and 19 test conversations as in the baseline approach (Stolcke et al., 2000; Kalchbrenner and Blunsom, 2013b). Table 4.6 shows the results of the proposed model with several setups, first without the context, then with one, two, and so on the number of the preceding utterances in context. We examined different values for the number of the hidden units of the RNN, empirically 64 was identified as best and used throughout the experiments. We also experimented with the various representations for the speaker ID that is

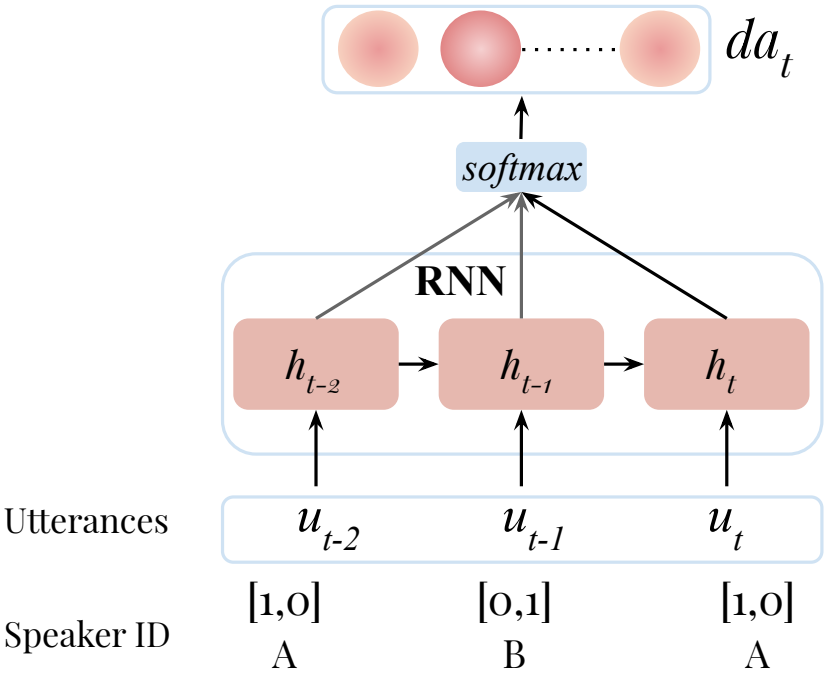


Figure 4.4: The RNN setup for learning the dialogue act recognition with the previous sentences as context and speaker identification. The utterance representation is concatenated with their corresponding speaker id.

concatenated with the respective utterances but could find no differences. As a result, our proposed model uses minimal information for the context. The performance increases from 74% without context to about 77% with context. We run each experiment ten times and take the average. When we vary the number of preceding utterances in the context, we run each experiment ten times and take the average. We discover that with three preceding utterances in the context, that is $n = 3$, the accuracy achieved is 77.34% with the standard deviation of 0.37 from mean over ten runs. The model shows robustness providing minimal variance, and using only three preceding utterances as a context can produce consistent results.

4.4.2 Analysis on Internal States of RNNs

We also analyze the internal state h_t of the RNNs for the two preceding utterances setup. We plot them on a 2D graph with the t-SNE algorithm for the first 2,000 utterances of the SwDA test set, as shown in Figure 4.5, the clusters of all the DA

Model setup	Accuracy
Baseline	
Most common class	31.50%
Utterance-level baseline model	73.96%
Related previous work	
Kalchbrenner and Blunsom (2013)	73.90%
Our work	
RNN (1 utt. in context w. SpeakerID)	76.48%
RNN (1 utt. in context)	76.57%
RNN (2 utts. in context)	76.81%
RNN (3 utts. in context)	**77.34%**
RNN (4 utts. in context)	77.28%

Table 4.6: Accuracy of the dialogue act recognition with the context-learning approach.

classes. The classes which do not share any information are grouped without any interference such as *Non-verbal*, and *Abandoned*. As we can also see in the figure that the big clusters belong to the dominating *Statement* classes, *sv* and *sd*. The *Question* classes, *qy*, *qw*, *qh* and *qo* are clustered within the big class. The classes *Backchannel*, *Yes-answers*, and *Agree/Accept* share much syntactic information; hence they are also clustered together, and our approach makes those classes separable within the cluster.

Figure 4.6 shows some particular classes with utterances in their vector spaces, the (1) current utterance and (2) a preceding utterance in the context. These are the examples of the context-sensitive dialogues, where we can see one cluster of the *ft* (Thanking) dialogue act class and three groups of the *fc* (Conventional Closing) dialogue act class. It is fascinating to notice that the phrases of the same dialogue act are clustered themselves in small chunks. For example, "talk you later" and "we'll talk again" are in a chunk close to each other. Whereas, the phrase "you too" created another chunk. Similarly, yet another small chunk with phrases "I appreciate" and "I sure enjoyed" are close to each other.

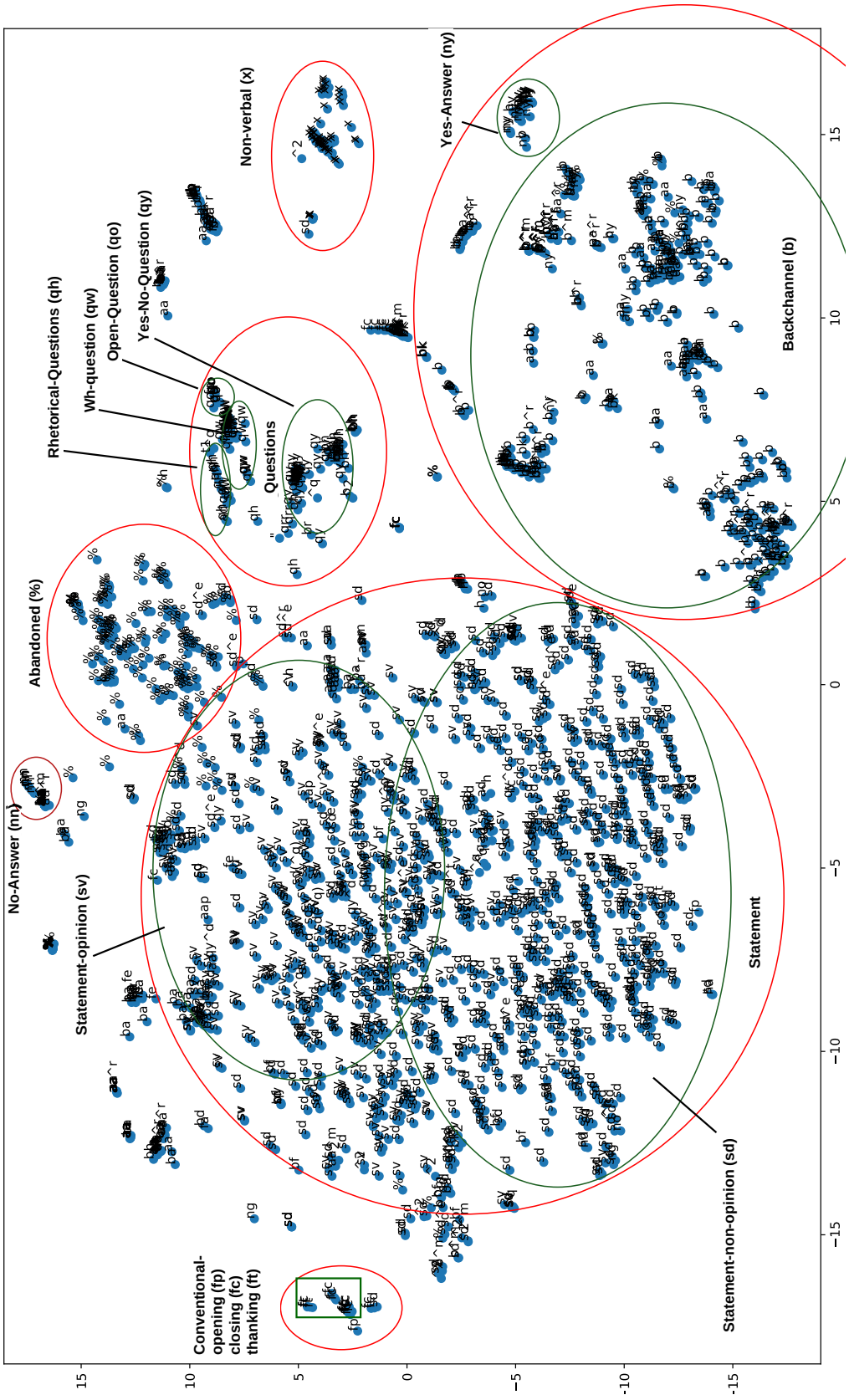


Figure 4.5: Clusters of all dialogue act classes in test set of SwDA corpus. For detailed tag set please visit Table B.1.

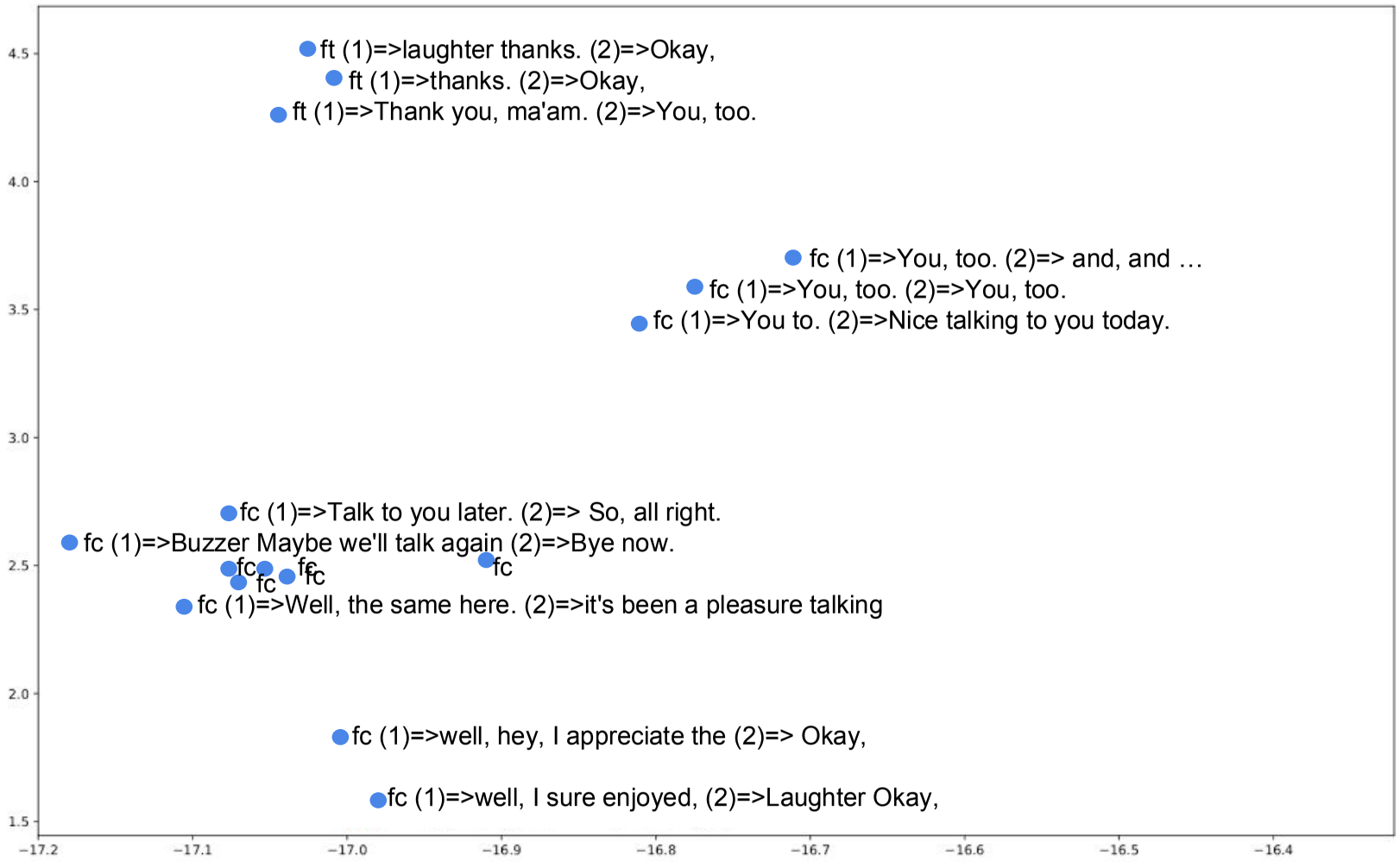


Figure 4.6: A blowup of the rectangle in Figure 4.5 from the Conventional Closing (*fc*) and Thanking (*ft*) DA classes with their utterances. For readability, some utterances have been omitted and we show only the labels.

4.5 Conversational Analysis using Utterance-Attention-BiRNN

Discourse analysis is a crucial task in the field of natural language processing; therefore, there exist many dialogue act corpora available (Serban et al., 2015), however, following the previous experiments, we use the Switchboard Dialogue Act (SwDA) corpus. We show various analytical observations on the SwDA corpus, which is annotated with the DAMSL tag set (Godfrey et al., 1992; Jurafsky et al., 1997). In this section, we will answer: How many preceding utterances are required in the context to recognize the DA of the current utterance sufficiently? Another critical answer we find out about How much each preceding utterance in the context contributes while recognizing the DA of the current one (Sbisà, 2002)? Utterance-level attention mechanism helps us model the DA recognition task to find out this phenomenon. Finally, we show that context-based models detect correctly and with higher confidence than the utterance-level model (especially in cases where context plays an important role). The Utt-Att-BiRNN model is shown in Figure 4.7, for which the main components are the bidirectional recurrent neural network (*BiRNN*) and *Attention* mechanism.

Bidirectional Recurrent Neural Network

BiRNN is an extended form of the unidirectional RNN (Elman, 1990), introducing one extra hidden layer (Graves et al., 2013; Schuster and Paliwal, 1997). The hidden to hidden layer connections flow into the opposite temporal direction, as discussed in Chapter 3. The model provides forward and backward states with corresponding directions of the hidden layers, as shown in Figure 4.7, and the final result which is a probability distribution over all dialogue acts (da) can be calculated as follows:

$$h_t^f = f\left(W_h^f h_{t-1}^f + W_u^f u_t + b_h^f\right) \tag{4.3}$$

$$h_t^b = f\left(W_h^b h_{t-1}^b + W_u^b u_t + b_h^b\right) \tag{4.4}$$

$$da_t|\{u_{final}(u_t, u_{t-1}, ...u_{t-n})\} = g\left(W_{da}^f h_t^f + W_{da}^b h_t^b + b_{da}\right) \tag{4.5}$$

where n is the number of preceding utterances in the context for time instance t, and u_{final} is calculated using attention mechanism discussed in the next section. W and h are the corresponding weight matrices and hidden vectors, where the

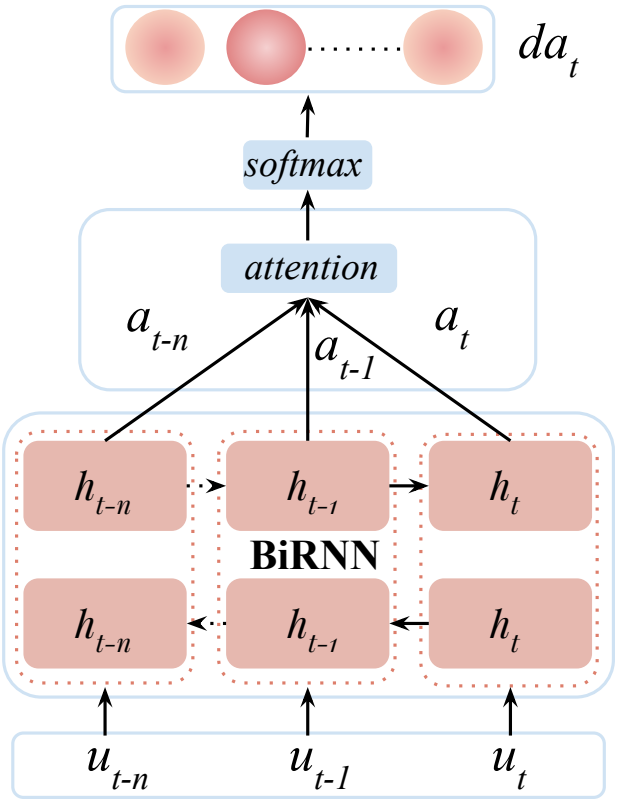


Figure 4.7: Utt-Att-BiRNN model for Dialogue Act Recognition.

superscripts f and b represent the forward and backward hidden layer directions respectively. In our scenario, we want the model to learn the context; thus, the input consists of the current utterance $(u_t)$and preceding utterances in the context $(u_t, u_{t-1}, ...u_{t-n})$. If we use the unidirectional RNN model, there might be a chance that the model becomes more attentive to the current utterance only. The sequential information in RNN is compressed to the final state as a hidden state (h_t). The bidirectional RNN model, on the other hand, exploits the information in all given input utterances by looking back and forth through them. Therefore, our goal is to treat all utterances equally and learn how much each utterance contributes to the final result.

Attention Mechanism

The attention mechanism is loosely based on visual attention found in humans, and broadly used in image recognition and object tracking, discussed in detail in Chapter 3 (Larochelle and Hinton, 2010; Denil et al., 2012). Nevertheless, recently,

attention mechanism with RNNs are being used for several natural language processing tasks, such as machine translation and comprehension, speech recognition (Bahdanau et al., 2015; Vinyals et al., 2015; Chorowski et al., 2015). We propose the attention mechanism to compute the contribution weights of the utterances for predicting the corresponding class. Given the number (n) of preceding utterances in an input sequence $u = \{u_t, u_{t-1}, ...u_{t-n}\}$, the BiRNN provides the respective hidden vectors $h = \{h_t, h_{t-1}, ...h_{t-n}\}$. The attention layer computes the weights $a = \{a_t, a_{t-1}, ...a_{t-n}\}$ as the contribution for every corresponding input utterance in u using the respective hidden representations h, as depicted in Attention part of Figure 4.7. Hence, the final utterance representation u_{final} of the utterance sequence in u is formed by a weighted sum of h and a:

$$m = tanh\left(W_h h\right) \tag{4.6}$$

$$a = softmax\left(W_m^T m\right) \tag{4.7}$$

$$u_{final} = tanh\left(ha^T\right) \tag{4.8}$$

where W is a trained parameter while W^T being its transpose. We use the $softmax$ function to compute the weights which provides $\sum_{n=0}^{n} a_{t-n} = 1$. It is important for the utterance-level attention mechanism that we normalize a to interpret the amount of contribution for each utterance in u.

Training Utt-Att-BiRNN Model

In the baseline model and the Utt-Att-BiRNN model settings, we use a $softmax$ function to predict a discrete set of classes da_t on top of the learned u_{final} representations. We use a set of 5 utterances in u, with the current utterance and four preceding utterances in the context. A similar study performed in (Bothe et al., 2018d) shows the effect of the number of utterances in the context. It was shown that three utterances provide sufficient context. However, we use four context-utterances to provide a large enough window for bidirectional exploration by the RNN, hence $n = 4$.

We minimize the categorical cross-entropy in all learning cases as we have multiple classes in the DA recognition task. We use 64 hidden units with the dropout regulariser (Hinton et al., 2012) in the BiRNN hidden layer for the proposed model. As a result, we get 128 hidden units as a concatenation of the h_t^f and h_t^b hidden units. These are the only parameters determined empirically for

Models	Accuracy
Prior related work	
Most common class baseline	31.50%
Our baseline model	73.96
Markov Model (Stolcke et al., 2000)	71.00%
C-RNN Model (Kalchbrenner and Blunsom, 2013b)	73.90%
Our work	
Character LM rep.	76.47%
Word-embeddings mean rep. (ConceptNet)	75.43%
Word-embeddings mean rep. (ELMo)	75.39%
Concatenated rep.	76.15%
Average char-word-level predictions	76.84%
Average char-word-level & concatenated rep. predictions	**77.42%**

Table 4.7: Accuracies on the SwDA test set of Utt-Att-BiRNN model with context.

the classification tasks, but all other parameters are learned during training. The Adam optimizer (Kingma and Ba, 2014) is used with an initial learning rate $1e$-4, which decays during training. Early stopping is used to avoid over-fitting the network, with 15% of the training samples for validation. We wait for at least five iterations over which the accuracy on the validation set does not improve.

4.5.1 Results with Utt-Att-BiRNN Model

The baseline and Utt-Att-BiRNN models are trained and tested using both the utterance representations explained in Section 4.3. We report the accuracies on the test set of SwDA corpus in Table 4.7. Character LM and word-embeddings mean utterance representations perform quite well for this task. Surprisingly, the word-embeddings mean representations of the utterances used from the ConceptNet seem to show fair results given the fact of the low dimensionality of the

GT	NC	WC	Num	pct.	Example of utterance
sv	*sd*	*sd*	198	4.73%	Uh, the problem is here
					But they don't have
					We're hearing the same
sd	*sv*	*sv*	51	1.22%	They're certainly legal,
					Real long legs,
					And time consuming,
aa	*b*	*b*	44	1.05%	Yes.
					Yeah.
					Uh-huh.

Table 4.8: The test samples from the SwDA corpus where both classifiers, simple utterance-level and Utt-Att-BiRNN, failed to correctly predict classes (the majority classes, Statement-non-opinion (*sd*) and Statement-opinion (*sv*), are reported here). Where **Num** is a number of samples, **GT** stands for ground truth, and **pct.** for percentage.

embedding vectors compared to character LM feature vectors. The word embedding vector has only 300 dimension size, whereas character LM feature vector has 4096 dimension size. However, the word vector might have out-of-vocabulary words that can be mitigated using a character-level language model. It can be seen in the results that the accuracy of the models is consistent with the character LM feature representations.

We also experiment with a combined model of these representations in two ways: first by concatenating both representations and using them as an input, and second by averaging both models' output predictions. Averaging the predictions has shown the best results that are trained with character LM, and word-embeddings mean vector representations. Concatenated representations deliver the best of the performance. We can see that context-based learning shows a performance improvement of about 4% (compared with utterance-level classification from Table 4.4).

GT	NC	WC	Num	pct.
ny	*b*	*ny*	33	0.79%
aa	*b*	*aa*	29	0.69%
aa	*sd*	*aa*	12	0.28%
b	*aa*	*b*	23	0.55%
b	*%*	*b*	16	0.38%

Table 4.9: The test samples from the SwDA corpus where the Utt-Att-BiRNN model correctly predict as opposed to the simple utterance-level classifier.

4.5.2 Analytical Examination on Failure of Recognition

We examined the SwDA corpus test set and found many of the instances wrongly predicted with both models. The dominant DA classes in the SwDA corpus are Statement-non-opinion (*sd*) and Statement-opinion (*sv*). Table 4.8 shows the number of samples (*Num*) and their percentage (*pct.*) out of 4,186 utterances. The examples of utterances present how difficult they are for humans to identify correctly. It shows ambiguity in two DA classes, *sd* and *sv*, which accounts for about 6% of accuracy reduction for both models. We also show the effectiveness of the pragmatic model, which predicts the correct class when the context is essential, see Table 4.9. For example, if the utterances like "Yes" or "Yeah" are followed by Yes-No Question (*qy*), the probability that the second utterance belongs to Yes-answer (*ny*) is higher than being in Backchannel (*b*) or Abandoned (*%*). Similar utterances to the *ny* class are used in the Agree/Accept (*aa*) class, but they are usually followed by *sv*, *sd*, *b*, or some other classes. In total, we found 330 samples which constitute around 7.88% of the samples that were correctly recognized by the Utt-Att-BiRNN model against the utterance-level model. It means that the context-model has clearly achieved about 8% higher accuracy than the utterance-level model.

4.5.3 Effectiveness of Context using Confidence Values

We also found that the prediction confidence of the Utt-Att-BiRNN model is higher than the utterance-level classifier. Figure 4.8(a) shows three rows for the

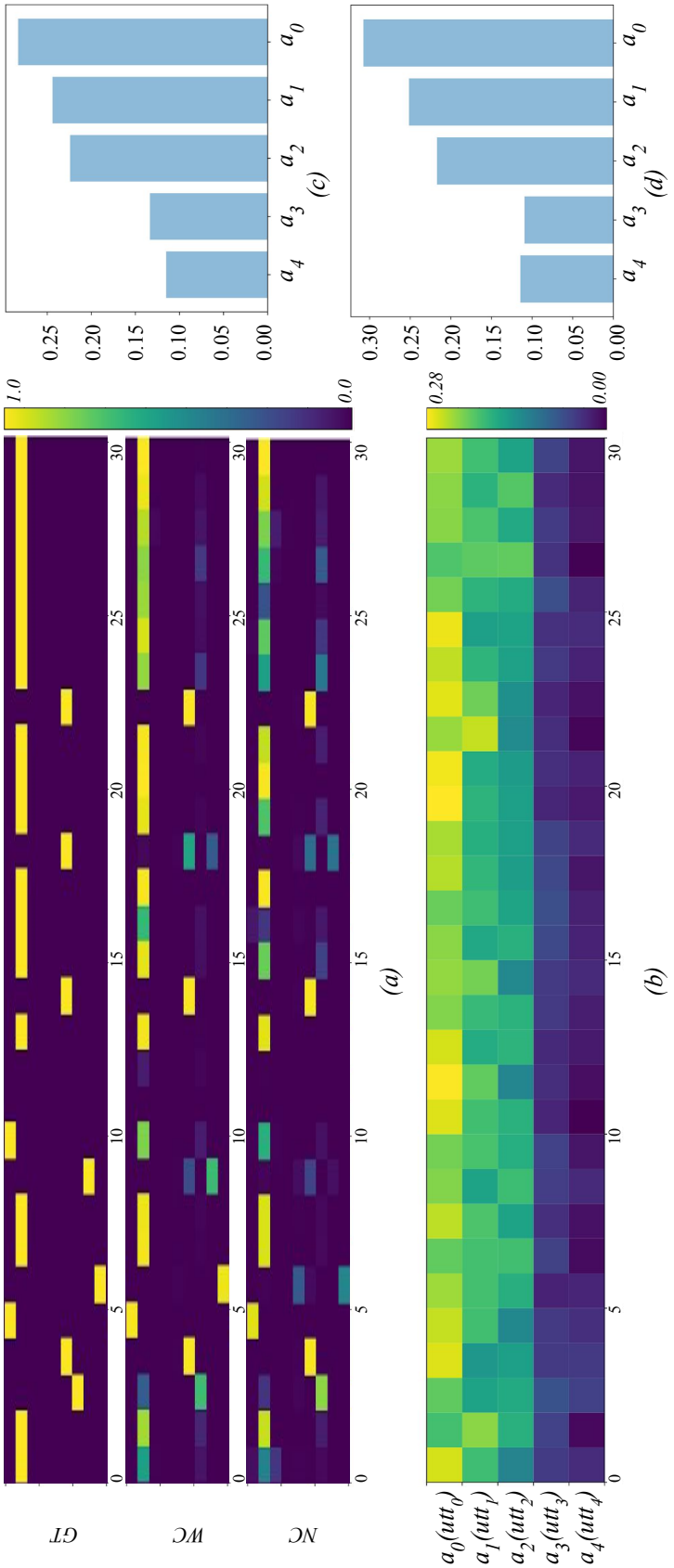


Figure 4.8: Effectiveness of the context. (a) Prediction confidence for a batch of 30 sets of utterances: the first row is the ground truth (GT), the second row the predictions with context (WC), and the third row the predictions with no context (NC). We show only 8 of the 42 classes for simplicity on the y-axis and the set of utterances on the x-axis. (b) The contribution of utterances $utt_0, utt_1, ...utt_4$ as the attention weights $a_0, a_1, ...a_4$. (c) The average weight of utterances and (d) in addition averaged over 10 runs to show robustness.

ID	Speaker	Utterances	GT	NC	WC
1	A	Okay, uh	*o*	*b*	–
2	A	could you tell me what you think contributes most to, uh, air pollution?	*qw*	*qy*	–
3	B	Well, it's hard to say.	*^h*	*fo*	–
4	B	I mean, while it's certainly the case that things like automobiles, factories, and active volcanoes.	*sv*	*sv*	–
5	B	What do you think?	*qo*	*qw*	*qo*
6	A	Um, well, you talked about, uh, volcanoes.	*sd*	*sd*	*sd*
7	A	I'm not sure how many active volcanoes there are now.	*sd*	*sd*	*sd*
8	A	I think probably the greatest cause is, uh, vehicles, especially around cities.	*sv*	*sv*	*sv*
9	B	Uh-huh.	*b*	*b*	*b*
10	A	Um, uh, do you live right in the city itself?	*qy*	*qy*	*qy*
11	B	No,	*nn*	*nn*	*nn*
12	B	I'm more out in the suburbs,	*sd*	*sd*	*sd*
13	B	but I certainly work near a city.	*sd*	*sd*	*sd*
14	A	Okay,	*bk*	*fc*	*bk*
15	A	so, can you notice...	*qy*	*sd*	*sd*
16	B	How about you?	*qo*	*qw*	*qo*
17	A	Well it's,	*%*	*%*	*%*
18	A	I live in a rural area.	*sd*	*sd*	*sd*
19	B	Uh-huh.	*b*	*b*	*b*
20	A	It's mainly farms and, uh, no heavy industry.	*sd*	*sd*	*sd*
21	A	Attleboro, itself, -	*sd*	*%*	*sd*
22	A	I live in Rhode Island.	*sd*	*sd*	*sd*
23	B	Oh, I see.	*b*	*bk*	*b*

Table 4.10: A piece of conversation from the test set of the SwDA corpus. Marked the utterances where with-context (WC) model outperformed over no-context (NC) model.

batch of 30 utterances in the DA recognition task: first ground truth (GT), second the predictions of the Utt-Att-BiRNN model (with-context model - WC), and third the predictions of the utterance-level classifier (no-context model - NC). The predictions of the Utt-Att-BiRNN model show higher confidence when compared to the predictions of the utterance-level model.

4.5.4 Contribution of Context Utterances using Attention Mechanism

We also computed the amount of contribution of the context utterances using the Utt-Att-BiRNN model. As discussed in Section 4.5, the attention weights $(a_t, a_{t-1}, ...a_{t-n})$ can be interpreted as the contribution of the utterances, as the u_{final} of the utterance sequence in u is formed by a weighted sum of h and a. Figure 4.8(b) shows the attention weights $(a_0, a_1, ...a_4)$ that represent the contribution of the corresponding utterances $(utt_0, utt_1, ...utt_4)$. The current utterance utt_0 contributes higher than others. However, the closest preceding utterances seem to contribute substantially, whereas the far preceding utterances also contribute with a little proportion. In Figure 4.8(c) and 4.8(d), we can see the average of the weights for the corresponding utterances.

The same piece of conversation is presented in Table 4.10 with the predictions from the no-context model and with-context model. The rectangle marks show the utterances where the with-context model correctly predicted the dialogue act against the no-context model. As it can be seen from Table 4.10, the context model accurately catches the minuscule differences such as Wh-questions (qw) and Open-questions (qo), in utterances 5 and 15. The question in utterances 5 and 16 starts with Wh-phrases ("What" and "How"). It is an indication that the no-context model predicts them as the Wh-question DA class. However, the with-context model predicts them as Open-questions as they are derived from their context in the conversation. We can look closely at the attention weights shown in Figure 4.9 (snippet of Figure 4.8(b)), which is for the set of utterances presented in Table 4.10. This figure shows the attention weights of the current utterance and the preceding utterances. We can see that when the utterance is a question (like qo and qy as in the utterance numbers 5, 10, 15, 16) the attention weight values are high. However, right after the question (for example, after qy in utterance 10), the answer gets lesser weight giving more attention to the question. The

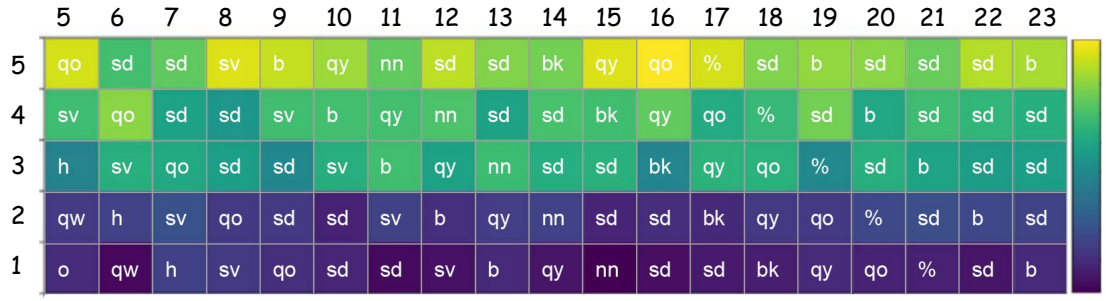


Figure 4.9: Attention weights of the utterances during dialogue act recognition for the conversation presented in Table 4.10. Note that the dialogue act of utterance number 5 is determined by using utterance 5 and preceding utterances 4, 3, 2, and 1. Similarly, DA of utterance 6 given utterances 6, 5, 4, 3, and 2.

utterance 11 with the No-answer (*nn*) dialogue act class, seem to be computed by giving more weight to utterance 10, which is preceding utterance with the Yes-No Question (*qy*) dialogue act.

Similarly, when the utterance dialogue act is a Backchannel (*b* or *bk*) like in utterance number 9, 14, 19, and 23, the model seems to take help from previous utterances. Typically, such Backchannel phrases are followed by statements (either Statement (*sd*) or Statment-opinion (*sv*) dialogue acts). Hence, it is natural to learn, for context-based models, the conversational behaviour to keep acknowledging the speaker with phrases like "Uh-huh", "Okay", or "Yeah" among others. Sometimes this can be tricky as the phrases like "Yeah" could also be followed by questions like a Yes-No Question (*qy*), in which case the dialogue act could be Yes-answer (*ny*). The phrases like "Uh-huh" or "Okay" could also be Accept/Agree (*aa*) dialogue act mostly in the context of Statment-opinion (*sv*). Hence, these utterances are contextually dependent as we explained early in this chapter and modelling them, in the same way, improves not only the performance but also aids the in-depth conversational analysis.

4.6 Summary

In this chapter, we explored one of the essential features of conversational analysis, the dialogue acts. We detailed the annotation and modelling of dialogue act corpora, and we highlighted that there is a difference in the way DAs are anno-

tated and how they are modelled. We argue to generalize the discourse modelling for conversation within the context of communication. Hence, we proposed to use the context-based learning approach for the DA identification task. In the first part, we used simple RNN to model the context of preceding utterances for the dialogue act of the current utterance. We used the domain-independent pre-trained character language model and word embeddings to represent the utterances. We evaluated the proposed model on the Switchboard Dialogue Act corpus and showed the results with and without context. For this corpus, our model achieved the accuracy of 77.34% with context compared to 73.96% without context. We also compared our model with Kalchbrenner and Blunsom (2013), who used the context-based learning approach similar to our method, achieving 73.9%. Our model used minimal information, such as the context of a few preceding utterances which can be adapted to an online learning tool such as a spoken dialogue system where one can naturally see the preceding utterances but not the future ones. It makes our model suitable for human-robot/computer interaction which can be easily integrated into any real-time spoken dialogue system. Our experiments answer a fundamental question, how many utterances in the context are contributing to the dialogue act recognition.

In the second part, we have presented the Utt-Att-BiRNN model for conversational analysis. We demonstrated that our model allows us to model context-based pragmatic learning and compute the amount of information used from the context utterances. This model also achieves a state-of-the-art result on the SwDA corpus of about 77% of accuracy, using only preceding utterances in the context. We showed that our model correctly predicted a significant number of the instances on the DA recognition task. We also showed that the context-based learning approach shows higher confidence in the classification task compared to simple utterance-level classification. We have investigated different aspects of the conversational analysis and showed that the proposed model could compute the contribution of the preceding utterances. The utterance-level attention mechanism also helped us determine how much (using attention weights) the preceding utterances actually contribute to the subsequent utterance. In this research, we only analyzed the utterance representations based on transcripts. However, we perceive that audio features could provide better representations for the utterances because of the change in sound intonation for the same utterance might be different with different dialogue acts. Furthermore, it would also help to an-

alyze and mitigate the influence of transcription errors. We investigated the DA annotations by reviewing the predictions of different models that could be extended to determine a reliable metric to accompany accuracy to assess the model performance.

Chapter 5

Emotion and Sentiment Analysis in Dialogues

Emotion or sentiment recognition plays an important role in natural language understanding and human-robot interaction to comprehend users feelings, unlike dialogue acts, provide meaning and semantic information. Emotional or sentimental expression are essential cues in the decision-making process during empathetic and affective dialogue. The emotion recognition becomes challenging when other modalities are absent. In this chapter, we will apply dialogue-based learning to emotion and sentiment analysis only with the textual conversation data to understand the affective context in the dialogues.

5.1 Introduction

Emotions are rich and crucial socio-linguistic cues in communication that have been subject to study for several years in the field of psychology, sociology, medicine, and computer science. As explained in the article, "Emotional Intelligence" (Salovey and Mayer, 1990): "emotions are viewed as organized responses, crossing the boundaries of many psychological subsystems, including the physiological, cognitive, motivational, and experiential systems". Furthermore, the common viewpoint drawn among many studies that the emotions are raised in a response to certain incidences (Salovey and Mayer, 1990; Ekman, 1992; Mundra et al., 2017; Gupta et al., 2017). Emotional intelligence is strongly related to social and communication intelligence (Gardner, 2011). An illustration of the emotion-driven contextual dialogue is shown in Figure 5.1, where the health assistant

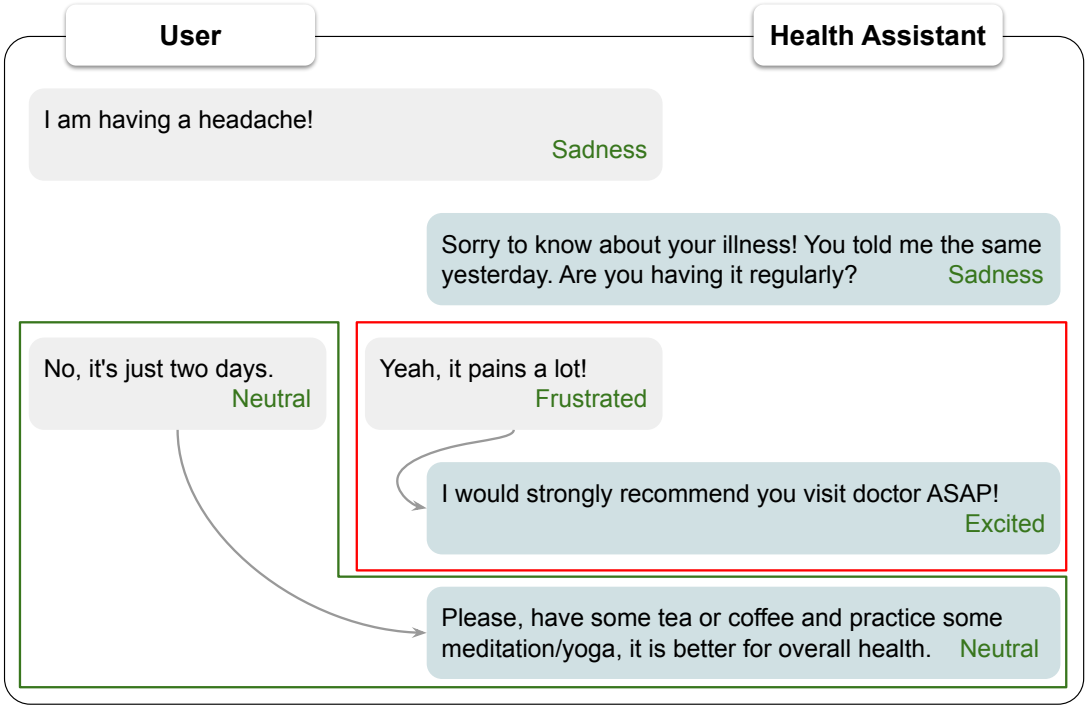


Figure 5.1: Illustration of an Emotion-driven Contextual Dialogue: different responses are produced for the perceived emotion from the user input and given the context information - excited response in an emergency situation (red box) and neutral response in a normal case (green box).

responds based on the emotion and context perceived from the users input, motivated by the example in (Poria et al., 2019). When the user suffered an illness for many days, the health assistant responds with *excited* emotion showing emergency (shown in the red box) otherwise responds with *neutral* emotion providing general health suggestions (shown in the green box).

Emotion detection in a text is a challenging task, especially when there is no other information available such as facial (visual information) expressions or prosodic (audio signals) features (Ekman, 1992). As emotion expressions are reactions to the incident or event that happened in the context, it is useful information to ease the emotion detection process. For example, as illustrated in Figure 5.1, the health assistant makes appropriate use of the context information provided in the users' utterances to respond accordingly to empathy. Similarly, when a person posts a thought (in the form of text) on social media, people respond to it based on the perceived emotion. The responses could then provide a proper setup for a few turn dialogues that can help perform contextual emotion detection. A

similar experiment is conducted in the EmoContext competition, where Tweets are used as a context. As the Tweet responses, posts are recorded to form the three-turn dialogues (Gupta et al., 2017).

In the next section (Section 5.2), we will explore an experiment where the Tweets are given with a goal to detect emotion intensity values of the Tweet messages, where the contexts are absent. This experiment is conducted in a challenge called EmoInt (Emotional Intensity Detection Challenge), which helps explore different language features for emotion detection, where we propose an ensemble model (Lakomkin et al., 2017). In the following section (Section 5.3), we use a similar setup of ensemble model, and apply it for the contextual emotion detection in dialogues (Bothe and Wermter, 2019). In this challenge, named EmoContext (Contextual Emotion Detection in Dialogue), the three-turn dialogues are given with a goal to detect emotion labels. The three-turn dialogues are treated similar to the dialogue act recognition task, as explored in the previous chapter, with two utterances in the context.

We further extend the idea of contextual emotion learning in dialogue to sentiment-guided learning to estimate the sentiment of the next upcoming utterance (Bothe et al., 2017), which is presented in Section 5.4. In this experiment, we find that humans use changes in a dialogue to specify or predict desirable and safety-critical situations and use them to react accordingly. The same cues can be used for safe human-robot interaction for early verbal detection of potentially dangerous situations. As a result, estimating the sentiment of the next upcoming utterance helps in inferring the situation from the preceding utterances in a conversation in the given scenario.

5.2 Emotion Intensity Detection from Tweets

The EmoInt shared task challenge has a goal to predict emotion intensity values of the Tweet messages. The text of Tweet and its emotion category (*anger*, *joy*, *fear*, and *sadness*) are given along with the intensity values. The participants were asked to build a system that assigns those emotion intensity values. Categorizing the emotion is already a challenging task, and emotion intensity estimation becomes an even more challenging problem. The main issues are the short length of the Tweet messages with the noisy structure of the text and the lack of sufficient annotated data. We developed an ensemble system of two neural models,

processing input on the character- and word-level, along with a lexicon-driven system. The correlation scores across all four emotions are averaged to determine the bottom-line competition metric. Our system ranked fourth place in full intensity range and third place in a 0.5-1 range of intensity among systems when writing the system description article (June 2017) (Lakomkin et al., 2017).

Introductory Background of Experiment

Sentiment analysis provides the degree of positive or negative of the opinion expressed by the user in the given text. Such information can be useful for providing better services for users (Kang and Park, 2014) or preventing potentially dangerous situations (O'Dea et al., 2015). However, the emotions (i.e. *anger*, *joy*, *fear*, and *sadness*) replace the traditional sentiment classes (such as positive or negative), and provide extra information on the opinion. On the contrary, a continuous intensity scale of emotion provides a fine-grained recognition of the emotion. The challenge in emotion recognition from the Tweet messages arises from several factors such as extensive usage of hashtags, slang, abbreviations, and emoticons.

The existing approaches, such as AffectiveTweets, heavily rely on manually constructed lexicons which contain information about intensity weights for each available word (Neviarouskaya et al., 2007; Mohammad and Bravo-Marquez, 2017a). The final intensity for the whole sentence is inferred by combining individual scores of the words. The main limitation of such models is ignoring word order or compositionality of the language that impacts sequence modelling. The deep learning approaches, such as recurrent neural networks, are deployed to learn the sequences and compositionality of the language for opinion mining (Irsoy and Cardie, 2014). Such data-driven approaches can overcome the limitations, and they have been powering many recent advances in natural language processing tasks, such as language modelling, machine translation, part-of-speech tagging, and classification (Biswas et al., 2015; Socher et al., 2013b; Radford et al., 2017; Irsoy and Cardie, 2014; Sailunaz et al., 2018).

This experiment augments the traditional lexicon-based models with two neural network-based models: character- and word-level inputs using recurrent neural networks. Character-level neural language models have shown promising results on natural language understanding tasks such as text classification (Zhang et al., 2015) and machine translation (Kalchbrenner et al., 2016). We use one of the character-level language model trained with the recurrent neural network to pre-

Split	***Joy***	***Anger***	***Fear***	***Sadness***	Total
Train	823	856	1147	786	3612
Validation	78	83	109	73	343
Test	714	760	995	673	3142
Total	1615	1699	2251	1532	7097

Table 5.1: EmoInt (Emotional Intensity Detection Challenge) dataset statistics.

dict the next character given the preceding ones. It is useful for a domain-specific text like Tweets where special kind of language features are frequently used such as hashtags, emoticons, or character repetitions. It also supports the intuition that a character-level model captures common writing patterns. A word-level recurrent neural model, on the other hand, can incorporate the order of word sequence using distributed embedding representation of words trained on a large amount of text data.

5.2.1 Ensemble Model for EmoInt

The input sentence goes through three models: AffectiveTweets model, character-level language model, and word-level embedding. AffectiveTweets model is a given baseline model that converts the input sentences into words and their respective lexical sentiment values. Character-level language model encodes each character of the input sentence, whereas the word-level model incorporates sequential information of the input text in the Tweet processed with the RNNs. Finally, the weighted average ensembles the output of all these models, the final ensemble model is shown in Figure 5.2. We will explain the model with the data preparation process of the Tweet messages.

Data and Preprocessing: The dataset statistics is shown in Table 5.1, it is comprised of a total of 7097 annotated Tweets, classified into 4 categories: *joy*, *anger*, *fear*, and *sadness* (Mohammad and Bravo-Marquez, 2017b). Each annotated Tweet is assigned with an ID, full text, emotion category, and emotion intensity value. Emotion intensity is a real value in the range from 0.000 to 1.000,

id	Sentence	Emo	Int
10005	My blood is boiling	anger	0.875
20258	Now #India is #afraid of #bad #terrorism.	fear	0.646
30010	I'm just still. So happy. A blast!	joy	0.917
30112	LOVE LOVE LOVE #smile #fun #relaxationiskey	joy	0.740
30328	@Rbrutti what a #happy looking #couple!	joy	0.542
40002	Feeling worthless as always	sadness	0.958

Table 5.2: Examples from the EmoInt dataset.

where a higher value corresponds to a greater degree of intensity of the given emotion label. The samples from the EmoInt corpus are presented in Table 5.2.

Intensifiers (such as 'really', 'too', 'so') increase the intensity of the given emotion, such as Tweets with id 30328 and 30010. On the other hand, repetitive use of words also intensifies the emotion class, like the LOVE word in 30112 is used to intensify *joy*. We can see that the data is quite unbalanced and contains non-useful information such as URLs and user mentions (@username). We strip out the URLs and user mentions, and keep only the following characters: a-zA-Z@-!:(),;?.#'0-9*. As a final preprocessing step, we always lowercase the Tweet text before processing with the neural models.

Baseline model: The baseline system is a WEKA-based model called AffectiveTweets (Mohammad and Bravo-Marquez, 2017a). This system combines features derived from several lexicons like MPQA (Wilson et al., 2005), Bing Liu (Hu and Liu, 2004), SentiWordNet (Baccianella et al., 2010), and others, more information in (Lakomkin et al., 2017). In addition, AffectiveTweets incorporates SentiStrength values (Thelwall et al., 2012) and Brown clusters (Brown et al., 1992) trained on ∼53 million Tweets[1]. They are combined with averaged and concatenated first k word embeddings of the Tweet. In the end, a Support Vector Machine algorithm is used as a regression model for predicting the emotion intensity values.

[1] http://www.cs.cmu.edu/~ark/TweetNLP/

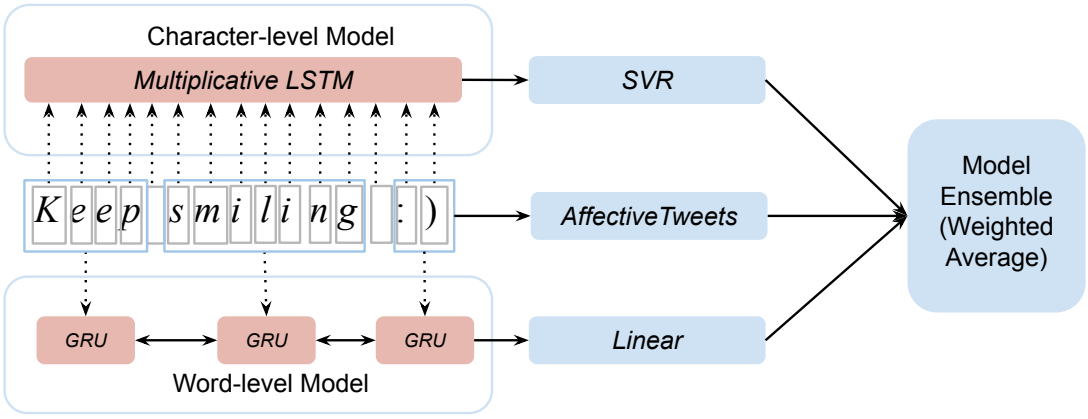


Figure 5.2: The ensemble model architecture for EmoInt Challenge. AffectiveTweets model with two neural models: character- and word-level models, and finally averaging the scores with weights tuned on the provided validation set.

Character-level RNN Model: We produce the character-level utterance representations as explained in Chapter 4, Section 4.3. In short, the character-level utterance representations are encoded with the pre-trained recurrent neural network model. This model contains a single multiplicative long short-term memory (mLSTM) network (Krause et al., 2016) layer with 4,096 hidden units, trained on ∼80 million Amazon product reviews as a character-level language model (Radford et al., 2017). The whole Tweet text is encoded with this pre-trained character-level language model. We encode text into a vector corresponding to the last character of a Tweet, average the representations of all hidden vectors and a concatenation of the two vectors. We also train the character-based language model on the Sentiment 140 corpus comprises of 1.6 million Tweets (Go et al., 2009). This model is small compared to the original pre-trained model, with only a single-layer LSTM (Hochreiter and Schmidhuber, 1997) of 1024 hidden units. It is trained using Adam optimizer (Kingma and Ba, 2014) with the learning rate of 0.0005 and clipping gradients at norm 1. We use the Support Vector Regressor (SVR) algorithm to classify Tweets represented as a fixed-length vector (of 1024 units) with the character-based recurrent neural network. Results reported in Table 5.3 show that the addition of the average vector to the features improves the model's overall performance. Surprisingly, the pre-trained character-level language model performed well among all the models; hence, this model produces the final results.

Range	**(0.0-1.0)**		**(0.5-1.0)**	
Model	**avg_p**	**avg_s**	**avg_p**	**avg_s**
PT, LV	0.470	0.468	0.412	0.404
PT, LV+AV	**0.474**	**0.472**	**0.419**	**0.413**
Twit, LV	0.312	0.307	0.296	0.288
Twit, LV+AV	0.319	0.310	0.298	0.301

Table 5.3: Effect on the avg_p (Pearson) and avg_s (Spearman) coefficients of different character language model (Char_LM) representations: last cell vector (LV) of the pre-trained model (PT, LV) and Twitter-specific character LM (Twit, LV). A concatenation of the last cell vector with the average of all cell vectors (AV) for the pre-trained model (PT, LV+AV) and Twitter model (Twit, LV+AV).

Word-level RNN Model: We use the distributed word embedding representations to encode words in the Tweet and experimented with different initialization methods. First, we randomly initialize the word embeddings and train the bidirectional gated recurrent unit (GRU, explained in Section 3.2.2) network (Chung et al., 2014). We set empirically 32-dimension cell size for modelling of the Tweet as a hidden memory vector in GRU. Then we replace the randomly initialized embeddings with two pre-trained versions of GloVe embeddings (Pennington et al., 2014) trained on Wikipedia and Twitter[2] to test if Twitter-specific word representations (domain-specific) are more suitable to solve the problem. Out-of-vocabulary words are replaced with a particular word 'OOV' and initialized as a random vector which is tuned during the training. We used a 50-dimensional embedding representation in all our experiments. GRUs are better in mitigating the vanishing gradient problem of the RNNs during the training and contain fewer parameters than LSTM units. The vector corresponding to the last word is fed to a fully connected layer with one neuron predicting emotion intensity. Results reported in Table 5.4 show that the Twitter GloVe embeddings could not show better improvement, whereas the Wiki GloVe embeddings outperformed the word-level models, which is used for the final results.

[2]`https://nlp.stanford.edu/projects/glove/`

Range	**(0.0-1.0)**		**(0.5-1.0)**	
Model	**avg_p**	**avg_s**	**avg_p**	**avg_s**
Random emb.	0.291	0.276	0.250	0.227
GloVe (Twitter)	0.300	0.293	0.231	0.220
GloVe (Wiki)	**0.326**	**0.323**	**0.259**	**0.252**

Table 5.4: Effect of different word embedding initialization techniques for the word-level model: randomly initialized embeddings, pre-trained GloVe embeddings on the Twitter and Wikipedia datasets.

Ensemble of Models: Ensembling different models is a widely used method to improve the performance of the system by combining the outputs of several classifiers. There are many ensembling techniques: mixing experts (Jacobs et al., 1991), model stacking, bagging and boosting (Breiman, 1996), and a simple weighted average of the scores of individual models, which is being used in this experiment. The small data size and use of the complex neural models might lead to overfitting; however, a simple weighted average of the scores of models lead to fair comparative results (López-Cózar et al., 2010). The final output emotion intensity value is calculated as a linear combination of individual predictions of three models (baseline, character- and word-level model):

$$emotion_{intensity} = w_b * baseline_{emotion} + w_w * w\_rnn_{emotion} + w_c * c\_rnn_{emotion} \quad (5.1)$$

such that:

$$w_b + w_w + w_c = 1 \quad (5.2)$$

where $baseline_{emotion}$, $w\_rnn_{emotion}$ and $c\_rnn_{emotion}$ are intensity predictions of the baseline, character- and word-level models corresponding to the emotions (*joy*, *anger*, *fear* or *sadness*). w_b, w_c and w_w are the ensembling coefficients, they were tuned on the given validation set to maximize the average Pearson correlation coefficient using grid-search.

5.2.2 Results and Discussion on EmoInt

The final Pearson and Spearman correlation score report of the experiments is presented in Table 5.5. These results were calculated through an online tool of

Range	**(0.0-1.0)**		**(0.5-1.0)**	
Model	**avg_p**	**avg_s**	**avg_p**	**avg_s**
Baseline	0.655	0.652	0.475	0.449
Char_LM	0.474	0.472	0.419	0.413
Word_Level	0.326	0.323	0.259	0.237
Char_LM + Word_Level	0.659	0.656	0.471	0.467
Char_LM + Word_Level + Baseline	**0.721**	**0.717**	**0.562**	**0.543**

Table 5.5: The final results of the ensemble model.

CodaLab, and they are available on the EmoInt Competition webpage[3]. We can see from the results that the ensemble model of the Char_LM, Word_Level, and Baseline model outperformed through the weighted average of the predictions. We can notice that Word_Level model alone could not achieve anywhere compare to Char_LM alone; however, ensembling it with Char_LM boosted the overall performance quickly. Also, given that these models are trained end-to-end without any external knowledge demonstrates the effectiveness of the character-level language modelling of noisy and short texts.

It is also worth noticing that the Char_LM and Word_Level models alone achieve lower correlation values than the Baseline model. That indicates that the Baseline model having external knowledge of the words (such as sentiment or semantics features through lexical models), helps to perform better than data-driven end-to-end neural models. However, they bring additional value to the ensemble model when added all together. The Tweet representations encoded with the pre-trained character language model obtained competitive results, and most surprisingly, our ensemble of Char_LM and Word_Level models alone also achieve better results than the baseline model. Our exploration of the feature representations for the utterance also shows that the average vector's addition could boost the results. Overall, we see that ensemble modelling and transfer learning helps in achieving state-of-the-art results.

[3] `https://competitions.codalab.org/competitions/16380#results`

5.3 Contextual Emotion Detection in Dialogue

When reading "I don't want to talk to you any more", we might interpret this sentence or utterance as either an angry or a sad emotion in the absence of context and other modalities. Often, the utterances are shorter, and given a short utterance like "Me too!", it is difficult to interpret the emotion without extra information. The lack of prosodic or visual information makes it challenging to detect such emotions only with single-turn text. However, using contextual information in the dialogue is important to provide a context-aware recognition of linguistic features such as emotion or sentiment and dialogue act. For this pilot study, we choose the SemEval 2019 Task 3 EmoContext competition that provides a dataset of three-turn dialogues each labeled with one of the three emotion classes, i.e. *Happy*, *Sad* and *Angry*, and in addition with *Others* as none of the aforementioned emotion classes. We develop an ensemble of the recurrent neural model with character- and word-level features to address the problem we explored in the previous experiment. The system performs quite well, and it ranked in the top 35% of the systems achieving a microaveraged F1 score (F1μ) of 0.7212 for the three emotion classes.

Introductory Background of Experiment

Humans might misinterpret the emotion in the text when reading sentences in the absence of context, so machines might too. When reading the following utterance,

```
Why don't you ever text me?
```

it is hard to interpret the emotion where it can be either a sad or an angry emotion (Chatterjee et al., 2019; Gupta et al., 2017). The problem becomes even harder when there are ambiguous utterances, for example, the following utterance:

```
Me too!
```

One cannot precisely interpret the emotion behind such an utterance in the absence of context. See Table 5.6 where the utterance "Me too!" is used in many emotional contexts such as *Sad*, *Angry*, and *Happy* and also in the class *"Others"* where none of aforementioned emotions is present. Analyzing the emotion or sentiment of the text provides the opinion cues expressed by the user. Such cues could help computers make better decisions to help users (Kang and Park,

id	turn1	turn2	turn3	label
2736	I don't hate you. you are just an AI	i don't hate anyone	me too	Angry
2867	everything is bad	whats bad?	me too	Sad
4756	I am very much happy :D	Thank you, I'm enjoying it :)	Me too	Happy
8731	How r uh	am fine dear and u?	Me too	Others

Table 5.6: Examples from training dataset, where *turn3* is mostly the same while emotional state is labeled differently, contextually.

2014) or prevent potentially dangerous situations (O'Dea et al., 2015; Sailunaz et al., 2018).

Usually, social media utterances are short and contain misspelt words, emoticons, and hashtags, especially in the textual conversation. Hence, using character-level language model representations can theoretically capture the impression of such texts. On the other hand, the EmoContext dataset is collected from the social media, and the character language model used in our experiments is also trained on a similar corpus of about ∼80 million samples (Radford et al., 2017). The hypothesis is that the character-level language model captures common writing patterns such as punctuation and signalling characters, for example, in "How r uh" shown in Table 5.6, character "r" signifies the word "are". In the absence of other modalities like vision or audio signals, the problem of detecting emotions becomes challenging while speaking those sentences. However, the given context of the utterances can help to mitigate the problem.

We propose a model that encapsulates character- and word-level features with recurrent and convolution neural network (Lakomkin et al., 2017). We use our recently developed models for the context-based dialogue act recognition, where we use a similar approach of the recurrent neural network and combine character language model and word embedding feature (Bothe et al., 2018b). Our final model for EmoContext is an ensemble average of the intermediate neural layers, ends with a fully connected layer to classify the contextual emotions. The

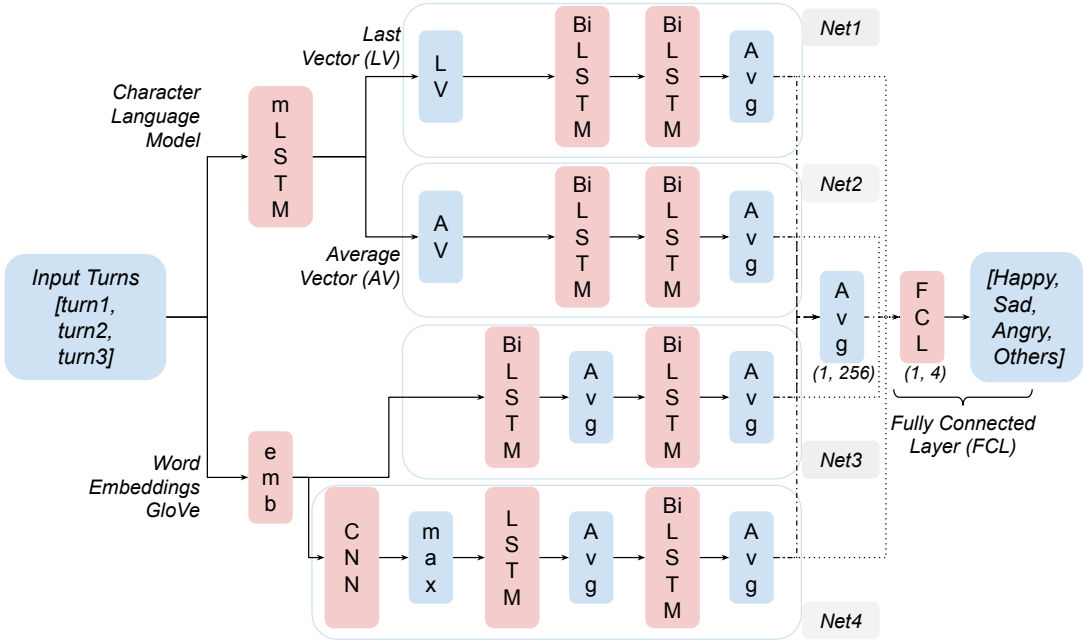


Figure 5.3: The overall ensemble model for the contextual emotion detection. This model ensembles four neural architectures: Net1 and Net2 - character LM last and average feature vectors with BiLSTMs, Net3 - uses word embeddings with BiLSTMs and Net4 - with convolutional neural network (CNN) with BiLSTMs. *max* represents the max pooling layer and *Avg* is the average layer.

system performance ranks in the top 35% (at the time of writing this article Feb 2019) on the public leaderboard at CodaLab Competition[4] achieving about 0.7212 microaveraged F1 score (F1μ) for the three emotion classes.

5.3.1 Ensemble Model for EmoContext

The final model used for submitting to the EmoContext challenge is shown in Figure 5.3. It is an average ensemble of four variants of neural networks. *Net1* and *Net2* use the input from a pre-trained character language model whereas *Net3* and *Net4* use the GloVe embeddings. We modified the previous architecture developed for EmoInt challenge explained in the previous experiment. There are significant changes in the current task of contextual emotion detection in the three-turn dialogues. The RNN is used to model the context of the two preceding utterances, similar to the dialogue act recognition task. We internally average

[4] `https://competitions.codalab.org/competitions/19790`

Split	***Happy***	***Sad***	***Angry***	***Others***	Total
Train	4243	5463	5506	14948	30160
Dev	142	125	150	2338	2755
Test	284	250	298	4677	5509
Total	4669	5838	5954	21963	38424

Table 5.7: EmoContext Data Distribution.

each network's output layers, as shown with a dashed line in Figure 5.3.

Data and Preprocessing: The dataset provided by the EmoContext organizers consists of the three-turn dialogues from Twitter, where *turn1* is a Tweet from user 1; *turn2* is a response from user 2 to that Tweet, and *turn3* is a back response to user 2 from user 1 (Gupta et al., 2017). The data statistics is presented in Table 5.7. We can see that the data is uniformly distributed over three classes (*Happy*, *Sad*, and *Angry*), but the class *Others* is dominated by a huge difference.

Character-level RNN Model: We use the same character-level utterance representations as in the previous experiment. It is a pre-trained recurrent neural network model which contains a single multiplicative long short-term memory (mLSTM) (Krause et al., 2016) layer with 4,096 hidden units (Radford et al., 2017). *Net1* and *Net2* are fed the last vector (LV) and the average vector (AV) of the mLSTM respectively. It is shown in (Lakomkin et al., 2017) that the AV contains compelling features for emotion detection. The character-level RNN models (*Net1* and *Net2*) are identical and consist of two stacked bidirectional LSTMs (BiLSTM) followed by an average layer over the sequences computed by last BiLSTM.

Word-level RNN and RCNN Model: The word embeddings are also used to encode the utterances. We use pre-trained GloVe embeddings trained on Twitter[5] with 200d embedding dimension (Pennington et al., 2014). The average length of

[5]`https://nlp.stanford.edu/projects/glove/`

Models	**F1μ**
Baseline model (organizers)	0.5838
Our proposed model	0.7212

Table 5.8: Results compared to other work; microaveraged F1 score (F1μ) for the three emotion classes, i.e. *Happy*, *Sad* and *Angry*.

the utterances is 4.88 (i.e. ∼5 words/utterance on average), and about 99.37% utterances are under or equal to 20 words. Therefore, we set 20 words as a maximum length of the utterances.

Net3 is stacked with two levels of BiLSTM plus the average layer, while *Net4* consists of a convolutional neural network (CNN). CNN in *Net4* over the embedding layer captures the essential features followed by a max-pooling layer (max), with the kernel size of 5 with 64 filters and all the kernel weights matrix initialized with Glorot uniform initializer (Glorot et al., 2011; Kim, 2014; Kalchbrenner and Blunsom, 2013b). The max-pooling layer of the size of 4 is used in this setup. This architecture eventually leads to building a recurrent convolutional neural network (RCNN) model by cascading the LSTM and then a stack of BiLSTM and the average layer to model the context.

Ensemble Model: As explained in Section 5.2.1, ensemble modelling is a widely used method to improve the performance of the system by combining the outputs of several classifiers. Among different ensembling techniques (mixing experts (Jacobs et al., 1991), model stacking, bagging and boosting (Breiman, 1996)), we use an average of the intermediate layers and the scores of individual models. The small data size and use of the complex neural models might lead to overfitting; however, a simple average of the models' representations lead to fair comparative results. The overall model is developed in such a way that the outputs of all the networks (*Net1*, *Net2*, *Net3*, and *Net4*) are averaged and the fully connected layer (FCL) is used with *softmax* function over the four given classes. The complete model is trained end-to-end so that given a set of three turns, the model classifies the emotion labels.

Models	**Accuracy**	**F1μ**
Char-LM AV Model (*No Context*)	86.26%	0.603
Char-LM LV Model (*Net1*)	88.12%	0.655
Char-LM AV Model (*Net2*)	90.25%	0.694
Word Embs Model (*Net3*)	88.27%	0.665
Word Embs Model (*Net4*)	88.80%	0.653
Char-LM Models (*Net1* and *Net2*)	89.59%	0.688
Word Embs Models (*Net3* and *Net4*)	87.91%	0.692
Average Ensemble Model *(avg. of outputs of individual networks)*	**91.31%**	**0.721**
Final Ensemble Model	**91.34%**	**0.721**

Table 5.9: Results comparing our experimental setups. The Char-LM AV features outperforms for No Context model as well as for the context model (Net2). We notice that the ensemble models performs very similar either ways: averaging the final outputs of the individual nets and final neural ensemble with FCL.

5.3.2 Results and Discussion on EmoContext

The final submitted result to the challenge is shown in Table 5.8. The metric used for the challenge is microaveraged F1 score (F1μ) for the three emotion classes, i.e. *Happy*, *Sad* and *Angry*. The EmoContext challenge organizers calculate it through the CodaLab online interface. Our model performance could compete quite well with the participating teams in the challenge. The main goal to present these experiments is to explore the features used for contextual emotion detection. To compare different language features (character and word) and neural network setups, we consider calculating the accuracy over all four classes alongside the score F1μ. The experimental setups of each network and ensemble models are tested individually; the results are reported in Table 5.9.

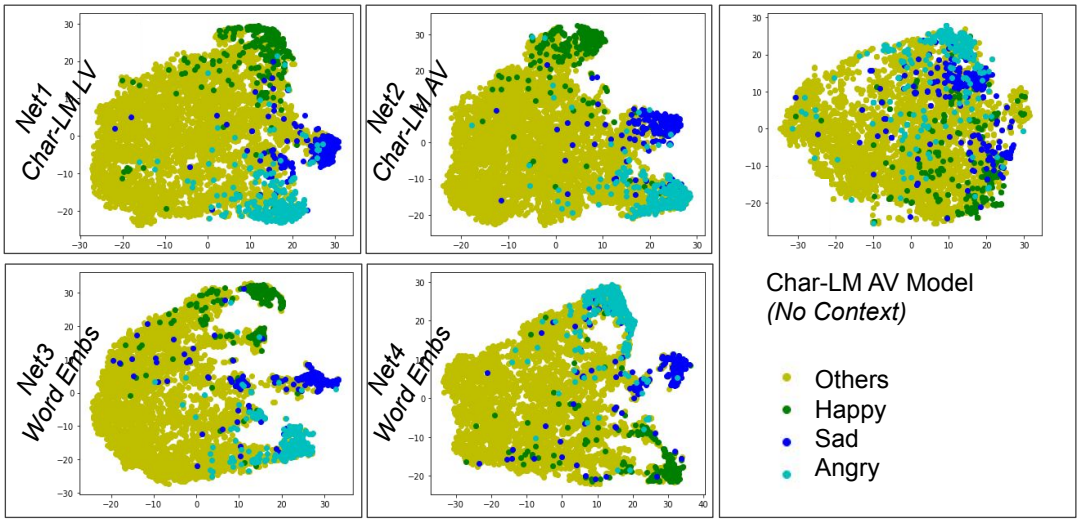
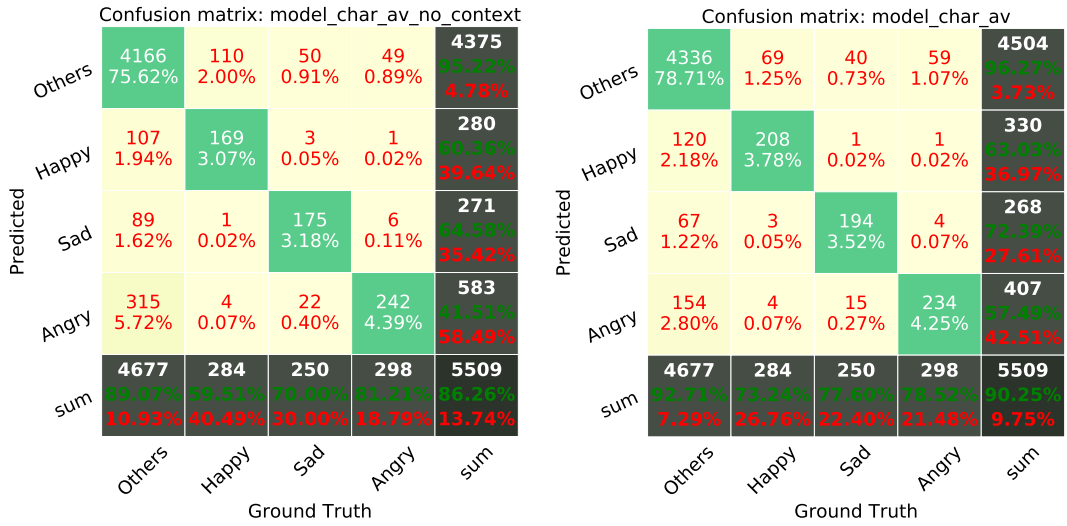


Figure 5.4: Clusters of the intermediate representations of individual networks on EmoContext test data. Legends given in this figure are applicable to Figure 5.6 and 5.7.

When the models train individually, the model's output is directly connected to the FCL shown by the dotted lines in Figure 5.3. The results show that the average vector Char-LM AV Model outperforms the four individual networks. As this model performs well, we also train a single FCL to see the effect of the absence of context. The ensemble models, Char-LM Models (*Net1* and *Net2*) and Word Embs Models (*Net3* and *Net4*) show a clearer raise in accuracy than individuals. The final ensemble model shows a definite improvement in the overall performance. However, we also ensemble the output predictions of all the individual networks, and average them at the end. Such an ensemble method is also useful for the overall improvement in the performance.

We took the intermediate representations at the last average layers of the networks on test data. We plot them against four given classes with the help of the t-SNE algorithm that converts multi-dimensional (256) array to 2-dimensional array. In Figure 5.4, we demonstrate the clustering of the individual networks. It also shows the individual network with no context model, where the model achieved 86.25% accuracy and 0.6 microaveraged F1 score (F1μ) for the character-LM AV utterance representation. We see four individual clusters of the network outputs, where the Net2 Char-LM AV model achieves higher isolation between the classes. Hence, it is worth experimenting with the same model without context



(a) No Context Model

(b) With-context Model

Figure 5.5: Confusion matrix of models with character-LM representation. No context model achieves relatively comparable performance to the with-context model, but we see that mostly the *Others* class is more confused than the emotion classes.

(Char-LM AV Model with No Context). We see that despite achieving competitive accuracy, the model fails to isolate the classes in the given space. Which indeed shows the effect of the contextual emotional dialogues that are learned with-context models better than no-context models, as can also be elicited from the confusion matrices given in Figure 5.5.

Figure 5.6, shows the clustering of the models comparing the average ensembles of character-LM representation models (*Net1* and *Net2*) and word-embedding representation models (*Net3* and *Net4*). From this figure, we discover that the character-LM representation models learn the isolation of classes better than the word embedding representation models. However, these ensemble models provide indifferent performance in terms of accuracy. Figure 5.7 shows clustering on the final ensemble models. We can notice that the *Net2* Char-LM AV model is entirely consistent while other models are a bit unstable in clustering for the given emotion classes. Surprisingly, for the final ensemble model, word models become too cluttered but still substantially contribute to the improvement. Figure 5.8 shows the confusion matrix for the final ensemble model where we can see numerically that the class *Others* is mostly confused. Whereas

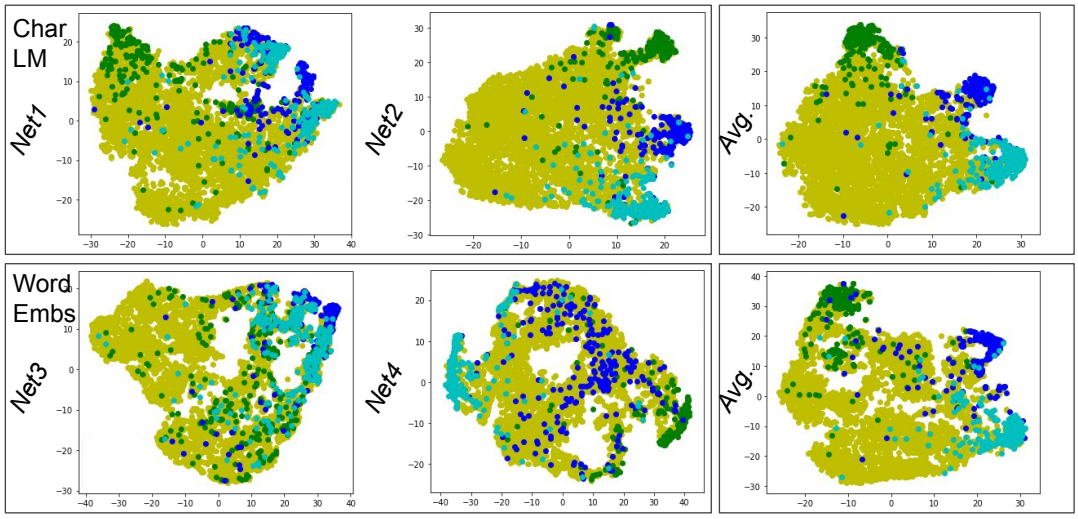


Figure 5.6: Clustering the intermediate representations of every two networks average ensemble on the EmoContext test data. The Net1 and Net2 outcomes are averaged (Avg.) to produce the final results, similarly Net3 and Net4.

the emotion classes *Happy*, *Sad* and *Angry* are clearly recognized as also seen how they are separated in the final cluster plot. Another critical point to notice from the confusion matrices is that the accuracy of both the ensemble models is very indifferent. That gives an intuition that averaging the final results or the

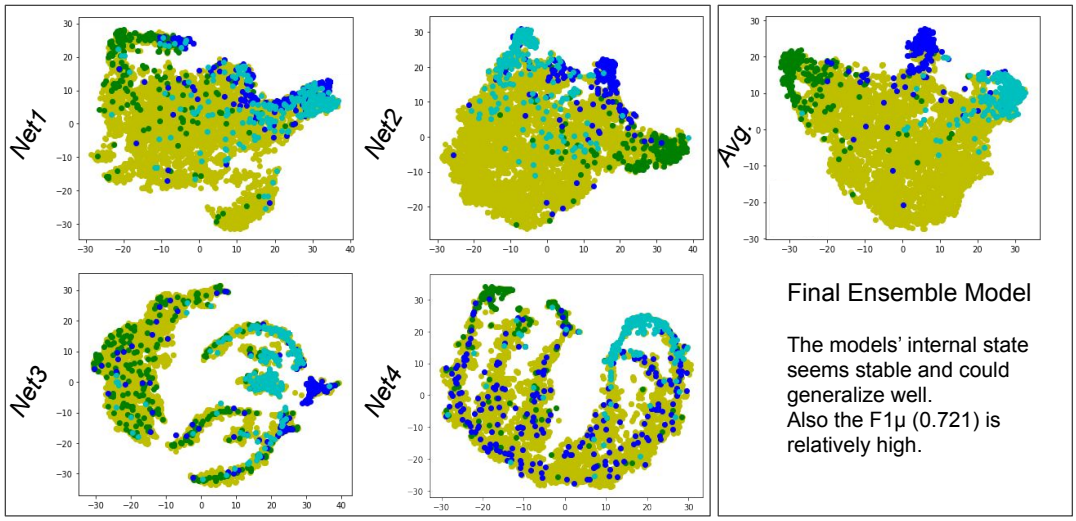
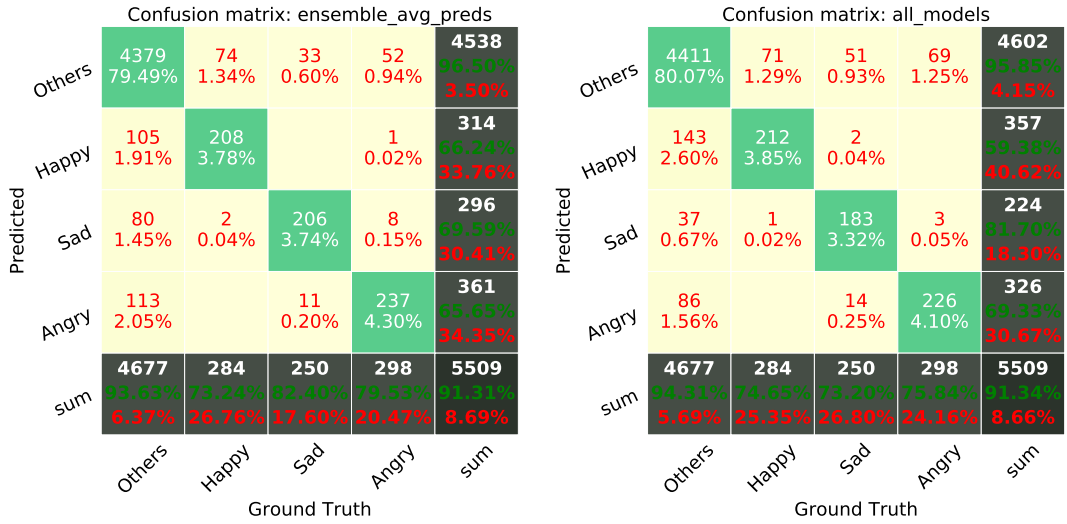


Figure 5.7: Clustering the intermediate representations final ensemble representations on the EmoContext test data. All networks are averaged (Avg.) to produce the final ensemble model.



(a) Ensemble Avg. Predictions

(b) Final Ensemble Model

Figure 5.8: Confusion matrix on the final ensemble with-context models. Both the ensemble models produce very similar results.

networks' intermediate representation produces nearly the same results.

Hence, to conclude the contextual emotion detection, it is essential to consider contextual neural modelling as a crucial step towards conversational analysis. Especially in the absence of other modalities such as facial expressions and prosodic features, context becomes an essential asset for emotion detection in the textual conversation. As we can see from the results, our model could compete and provide insight to explore different feature representations. The ensemble modelling and transfer learning become the practical tools for such a challenging task, specifically, when the given data is small, and the labels are not balanced over all the samples.

5.4 Sentiment-guided Dialogue-based Learning

In a conversation, humans use changes in a dialogue to predict safety-critical situations and use them to react accordingly. In this experiment, we propose to use the same cues towards safe human-robot interaction for early verbal detection of dangerous situations. We use a sentiment classifier to annotate the utterances of the targeted dataset due to no availability of the sentiment-annotated dialogue corpus at the time of this study. The goal is to learn the sentiment changes within the dialogues neurally and ultimately predict the sentiment of the upcoming utterance. We train the recurrent neural network on the context of word sequences from the two utterances of each speaker, to predict the sentiment class of the next utterance. Our results show that this leads to a useful estimation of the sentiment class of the upcoming utterance that can be used for early verbal detection of the potential safety-critical situations.

Introductory Background of Experiment

In human-robot interaction, one of the primary concerns is safety. In this work, we address safety as the condition of being protected from or unlikely to cause danger or injury. A mobile robot serving a wrong drink, coffee instead of water in a cup might be an acceptable mistake, whereas serving any drink in a broken cup becomes an unacceptable risk. When the robot is verbally instructed to perform this action, consider that the user also informs the robot that there is a chance of hazardous or risky situation, as illustrated in Figure 5.9. Early recognition of hazards is crucial for safety-related control systems, such as protective or emergency stop, which is an essential feature for personal care robots (Tadele et al., 2014). The main goal of this experiment is to study the early detection of safety-related cues through language processing. In the case of wrong robot action, the user might prompt with an utterance that, although often not understandable for the robot, carries a feedback signal for the last action performed, which can help understand the situation (Latham, 1997; Weston, 2016).

A possible conversation scenario as shown in Figure 5.9, the robot (R) perceives a sentence from the person (P) with neutral sentiment and responds with a query whether this means it should continue. Expecting a favourable (sentimentally positive) reply if everything is fine, but the next utterance has a negative sentiment. The robot can stop or revert the action based on this sentiment sig-

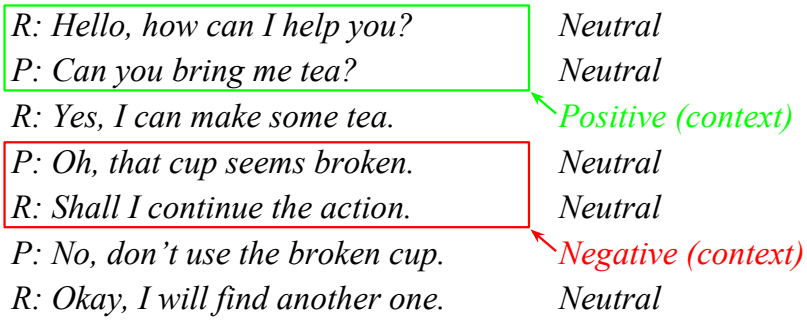


Figure 5.9: Example for preparing the context samples: labeled by sentiment analyser, previous two utterances of the sentiment classes (such as positive and negative) are stored as the context samples.

nal without understanding the utterance. Furthermore, an estimate of the users' response sensitivity is necessary when the robot needs to ask safety-critical questions (Fong et al., 2003b).

The goal of this experiment, as a first step, is to learn from the spoken language dialogues to predict the sentiment of the next upcoming utterance that eventually leads to in-depth language learning of the safety-critical cues. As shown in Figure 5.9, we use two utterances as context, capturing a sequence with both speakers, to predict the next utterance sentiment from the first speaker. We deploy long short-term memory (LSTM) network to learn the sentiment change in the dialogues. Since we want to extend our model to longer contexts, we choose the LSTM RNNs and show that they could successfully learn to estimate the sentiment of the next upcoming utterance.

5.4.1 Background: Language Learning through Feedback

Responses from humans in the conversational interaction have been used in various ways in the human-robot scenarios. In student/teacher learning scenarios, to facilitate learning, a teacher gives positive and negative feedback depending on the success of the student (Latham, 1997). Weston (2016) has shown that the positive-negative sentiment in the teacher's response helps to guide the learning process. Other work (Sordoni et al., 2015) describes context-sensitive response generation in the field of language understanding and generation. They report that the model lacks in reflecting the agents intent and maintaining the consistency with the sentiment polarity. That means that the predictable changes in

the sentiment polarity may act as cues for the changing situations so sentiment change over the dialogue can be used as a feedback signal to learn the changes in the perceivable environment.

Sentiment analysis is a crucial aspect of the decision-making process and has received much attention in the scientific community (Pang and Lee, 2008). With the vast amount of data available for analysis, many methods have been explored (Kim and Hovy, 2004; Wang and Manning, 2012) recently. Deep learning has given rise to some new methods for the sentiment analysis task, outperforming traditional methods (Socher et al., 2013b; Dai and Le, 2015). Different NLP tasks can be performed independently using deep neural networks (Collobert and Weston, 2008). Especially in text classification, the advanced neural network approaches are used, such as convolutional neural networks (Kim, 2014) and recursive and recurrent neural networks (Biswas et al., 2015; Socher et al., 2013b). A fixed-size context window can be used to solve the variable length of language text sequences, but this fails to capture the dependencies longer than the window size.

The accessibility to large unlabelled text data can be utilized to learn word2vec model (Mikolov et al., 2013a), which attempts to encode the meaning of words and the structure of sentences. The learned word embeddings are then used for creating lexicons and have a reduced dimensionality compared to traditional methods. This approach has also been used for learning sentiment-specific word embeddings for sentiment analysis (Maas et al., 2011). Our approach utilizes such word embeddings to process by the LSTM network in order to learn the sentiment changes in dialogues.

5.4.2 Contextual Sentiment Learning of Next Utterance

Data and preprocessing: We have used two spoken interaction conversational corpora for training our model from two very different sources, child-adult interaction and movie subtitles. The first has the child-level language component taken from the TalkBank system, called CHILDES[6] (MacWhinney, 1991), where many child and adult speakers converse on daily issues. In this dataset, we selected the conversations with children of age 12 and above, which have sufficient verbal interaction capabilities and comparatively less grammatical mistakes

[6] `http://childes.talkbank.org` or `http://childes.psy.cmu.edu`

Datasets	CHI	MDC
Raw utterances	11.1k	304k
Contexts (neg-pos)	4.1k	189k
Contexts (neg-neu-pos)	6.2k	283k

Table 5.10: Dataset statistics for sentiment-guided dialogue learning.

(Clark, 1978). The other corpus is the Cornell Movie-Dialogues corpus (Danescu-Niculescu-Mizil and Lee, 2011), which is more structured, it is grammatically more correct, and is also larger than the child-interaction corpus.

As our goal is to predict the sentiment from a context, as shown in Figure 5.9, we need to annotate the utterances with the sentiment labels. The child-interaction corpus (CHI) already has word-level sentiment annotation, while the movie dialogues corpus (MDC) has none. We thus used the Vader sentiment analysis tool (Hutto and Gilbert, 2014) from the Natural Language Tool Kit (NLTK) (Loper and Bird, 2002) library to annotate each utterance with a determined sentiment polarity. We empirically adjusted the sentiment level threshold to 0.2 and 0.6 on the scale of 0 to 1 for both positive and negative classes to avoid imbalanced classes in our data. Data samples are extracted by selecting the utterance with the given sentiment label as a ground-truth and capturing the previous two utterances as a context sample. We have created two datasets for the experiment, creating contexts from utterances with a set of either binary negative/-positive (neg-pos) or multi-class negative/neutral/positive (neg-neu-pos) classes. The dataset details are shown in Table 5.10. While taking the previous utterances for each sample, we encounter the overlapping of utterances in the contexts, i.e. one utterance may appear in two contexts. The proposed classification model can operate on binary (neg-pos) and multi-class (neg-neu-pos) dataset.

Model: We used the well-established recurrent long short-term memory (LSTM) neural network (Hochreiter and Schmidhuber, 1997), a particular form of the recurrent neural network, discussed in Chapter 3 and shown in Figure 3.4. The sequence of the words is represented by their numeric indices in a dictionary to be proposed with the embedding layer, which is implemented as a standard MLP layer, as shown in Figure 5.10. The embedding layer randomly initializes the normalized vectors or can utilize already pre-trained embeddings, to represent

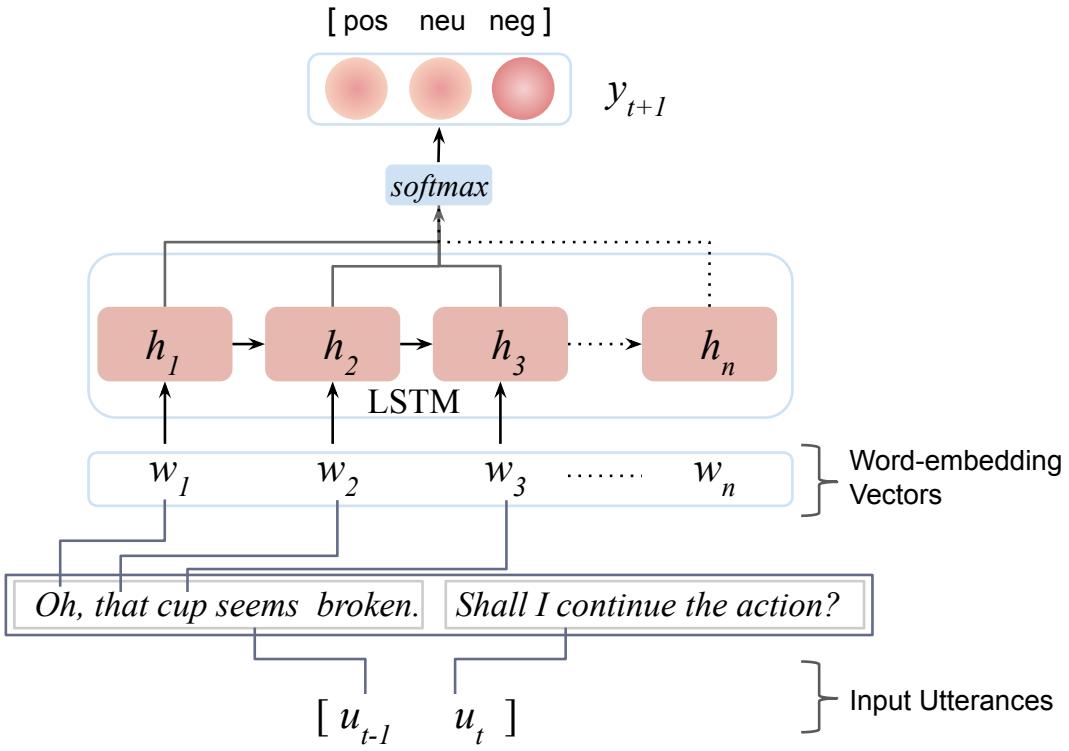


Figure 5.10: The long short-term memory (LSTM) units with classification setup. Biases are ignored for simplicity.

each word index by a real-valued vector of a given size which is then processed by the LSTM layer.

The LSTM layer receives a sequence of embedded word vectors $(w_1, w_2, ..., w_n)$ from the context utterances (u_{t-1} and u_t) as an input and outputs a sentiment class of the next utterance y_{t+1}. The last LSTM unit maintains a hidden vector h_n and different gates and memory cells responsible for controlling state updating at time step t as given in Equations from 3.5 to 3.10, as described in Section 3.2.2. The weight-projection matrices and bias vectors are initialized randomly and learned during the training process. The gating functions of the LSTM helps this RNN to mitigate the vanishing and exploding gradient problems. As an output, we get the hidden vector representation (h) of the entire sequence of words which is then used as an input to the $softmax$ classifier. In the classification setup as shown in Figure 5.10, given the current activation function in the hidden state h_t, the LSTM generates the output according to the following equation:

$$h_t = LSTM\,(w_1, w_2, w_3...) \mid \sum w \in (u_{t-1}, u_t) \tag{5.3}$$

Setups	Random Guess	CHI (10d)	CHI (100d)	MDC (100d)
Binary	50.00%	59.30%	59.06%	52.44%
Multi-class	33.33%	54.60%	54.56%	48.36%

Table 5.11: Prediction accuracy on the test dataset of the model trained with word embedding vectors.

$$y_{t+1} = softmax\left(W_{out} * h_t\right) \tag{5.4}$$

where W_{out} is an output weight matrix which can be stored to make the predictions along with other parameters as explained in Section 3.2.2. The *softmax* function produces normalized probability distribution over the possible classes, *positive*, *neutral* and *negative*.

5.4.3 Experiments and Results

The model is trained to recognize the sentiment polarities of the next upcoming utterance, given the recent two utterances in the context. We train the classifier by concatenating the context utterances and using the next utterance label as a training target signal (y_{t+1}). The utterances have been labelled by the sentiment analyzer for the binary and multi-class datasets, as given in Table 5.10. The input to the network is always the sequence of concatenated utterances. The prediction of the upcoming utterance sentiment is taken from the classified output of LSTM at the end of the sequence. The input sequence length is fixed to the maximum length in the utterances, and padding is used to make them of the same length.

The training is performed using categorical cross-entropy as the loss function, using the stochastic gradient descent optimization method. The learning rate and the number of hidden units were empirically determined for all the experimental setups. The hidden layer dimensions used are 64 for CHILDES and 512 for Movie-Dialogues corpus. We randomly initialized the word embedding vectors with the dimension of 10 and 100 for CHILDES and 100 for MDC, and we also used the pre-trained GloVe vectors of dimension 100 (Pennington et al., 2014). We trained the model on both the datasets as described before and for every two different setups. Each dataset is split into training, validation and test data with a 60%-

Setups	Random Guess	CHI (100d)	MDC (100d)
Binary	50.00%	63.36%	54.97%
Multi-class	33.33%	58.13%	51.71%

Table 5.12: Prediction accuracy on test data with the pre-trained GloVe word embedding vectors.

20%-20% split. The summary of the test data prediction accuracies is shown below in Table 5.11 where the word embedding vectors are learned by the model and 5.12 where pre-trained word embedding vectors are used. The pre-trained embedding representations show better accuracy than the randomly initialized representations, also, using different embedding dimensions (10 or 100) produced very similar results.

We also implemented a simple bot, that receives the utterances sequentially, evaluates the trained model on dialogue, and monitors the changing hypothesis of the upcoming utterances' sentiment. We present an example from the test data in Figure 5.11. The utterances from the conversation are processed one by one, and the progression of sentences is shown with the predictions and ground-truths. Bold values in the array [neg neu pos] represent the detected class for the current and the next utterance's sentiment hypothesis. We also show two related contexts, positive (green) and negative (red). For example, the utterance *"oh no, yeah this chair is broken"* has a negative sentiment label, and the model estimates a correct prediction. We can also see that the model failed to predict the positive class for the utterance *"yeah please use another one"*.

We notice an unpredicted increase in the negative sentiment for the utterance *"oh that chair is broken"*. However, the final result is still classified as neutral (towards negative), which could already have been used to detect a change in the sentiment. Thus the robot could be aware of a possible safety-critical change in the environment situation or the users' perception of the robots current action. The same can be noticed for the misclassified utterance where P2 perceived a negative situation and might have no solution. However, suddenly, interpreting the positive sentiment of P1 in the next utterance to understand that the situation has a solution or has been solved and no hazardous situation to expect. Overall,

	Utterances	*Sentiment of current utterance* [neg neu pos]	*Next utterance sentiment hypothesis* [neg neu pos]	*Next utterance might be*
	P1: ***please sit down***	[0.00 0.46 **0.54**]	[0.45 0.04 **0.51**]	***Positive***
Negative (context)	P2: ***yeah thanks***	[0.00 0.00 **1.00**]	[0.09 **0.78** 0.13]	***Neutral***
Negative (context)	P1: ***oh that chair is broken***	[0.44 **0.56** 0.00]	[**0.58** 0.20 0.22]	***Negative***
	P2: ***oh no , yeah this chair is broken***	[**0.46** 0.34 0.20]	[0.03 **0.94** 0.03]	***Neutral****
Positive (context)	P1: ***yeah please use another one***	[0.00 0.40 **0.60**]*	[0.28 0.09 **0.63**]	***Positive***
Positive (context)	P2: ***okay thank you***	[0.00 0.18 **0.82**]	[0.22 **0.59** 0.19]	***Neutral***

Figure 5.11: Test example: prediction on a dialogue.
* indicates that the sentiment recognition does not match the ground-truth.

the results show that it is possible to derive valuable cues by estimating the sentiment of the next upcoming utterance, and the model can learn to keep track of the sentiment through dialogues. The corpora used in this experiment, are auto-annotated with the standard sentiment analysis tool, which led to comprehensible results. However, a human-annotated corpus might still lead to better results.

Concluding Remarks and Discussion

In this experiment, we have presented a learning approach to estimate the sentiment of the next upcoming utterance within the dialogue. We show that the model can predict the sentiment of the upcoming utterance to a certain degree, taking into account that the used corpora are noisy. It is also important to mention that no system would reliably predict the upcoming utterance sentiments due to the changing nature of social dialogues. Detecting safety-related cues as early as possible is crucial, and a certain number of false-positives can be accepted (or quickly resolved through a query within the dialogue) if possible dangers can be avoided when they occur. We find that tracking even a noisy sentiment through the dialogue can positively impact safety during human-robot interaction, especially when combined with a multi-modal system.

While this work focuses on keeping track of the sentiment in dialogue-based context learning, we aim to extend this to different language features containing safety-related cues. The experiments show that the models can learn from the auto-annotated sentiment datasets. However, human-annotated labels might

lead to better results. This work already presents a promising step towards learning social cues via sentiment and can provide useful dialogue-based information regarding the safety context in human-robot interaction. However, we aim to explore different socio-linguistic features that contribute to more aspects of the language.

5.5 Summary

In this chapter, we explored one of the key socio-linguistic features, emotion. We provide an insight into how emotion influences the dialogue and decision-making process in general. We explored emotion intensity detection from the Tweets, where we learn how different features derived from several domain-specific lexica aids end-to-end learning of emotion detection. However, we found that RNN-based ensemble models alone could compete with the baseline model. Then we explored the contextual emotion detection in the data of three-turn dialogues, where we employ only RNN- and CNN-based ensemble models that competed for the state-of-the-art results. In the following section (5.3 "Contextual Emotion Detection in Dialogue"), we showed that the context learning of emotion elicits improved performance over the no-context model. In the last section (5.4 "Sentiment-guided Dialogue-based Learning") of this chapter, we presented a novel technique where the model is trained to estimate sentiment of next upcoming utterance using context-based learning. This technique helps determine undesired or dangerous cues through language learning in the conversation for safe human-robot interaction.

We aimed to use minimal information, such as transcribed textual conversations, which could be extended to different modalities such as prosodic features, as human voice changes in different emotional situations (Lakomkin et al., 2018). Sometimes such information of other modalities is entirely unavailable, such as on the social media and conversational chat. In such cases, the minimal textual information has to be utilized to achieve the ultimate task of emotion recognition. In contrast, while dialogue acts extract the meaning of an utterance in the dialogue, the emotions express feelings. In the next chapter (6 "Emotional Dialogue Acts"), we will show how emotion and dialogue acts posses some special relations. On the other hand, other socio-linguistic features, such as politeness, allow exploring an extended dimension of the emotion or sentiment. For example, polite behaviour

is seen with positive sentiment or happy mood emotion, whereas impolite is perceived as negative, aggressive, or unhappy with emotion (Langlotz and Locher, 2017). We will discuss about this concept in the last chapter (7 "Dialogue-based Navigation driven by Politeness for HRI"), where we demonstrate a human-robot interaction scenario that uses politeness cues in verbal interaction to vary the navigation speed of the robot.

Chapter 6

Emotional Dialogue Acts

We have explored dialogue acts and emotions independently; however, we discover that there is a strong relation between them, which can be useful to understand a different aspect of language. In this chapter, we present an ensemble of neural annotators to annotate existing emotion conversational data with the dialogue act labels and explore the discovery of the emotional dialogue act relationships.

6.1 Introduction

Emotion makes us understand feelings, whereas dialogue acts reflect the intentions and performative functions in the utterances. The recognition of emotion and dialogue acts can enrich conversational analysis and help build a natural dialogue system. It is quite evident from our conversational experience that when a person apologies (an Apology dialogue act) the expressed emotion is mostly sadness as against when thanking (a Thanking dialogue act), the emotion expression is mostly joyful, as illustrated in Figure 6.1. We aim to analyze such relations, but there was no conversational dataset available with the emotion and dialogue act labels together during this study. Most of the textual and multi-modal conversational emotion datasets contain only emotion labels but not dialogue act labels. To address this problem, we propose to use a pool of various recurrent neural network models trained on the Switchboard Dialogue Act (SwDA) corpus, in different architectural setups such as with or without context, as discussed in Chapter 4. We developed an ensemble of such neural annotators to annotate the emotion dataset for dialogue acts explained in this chapter. Each neural models annotate the emotion corpus with dialogue act labels, and an ensemble annotator

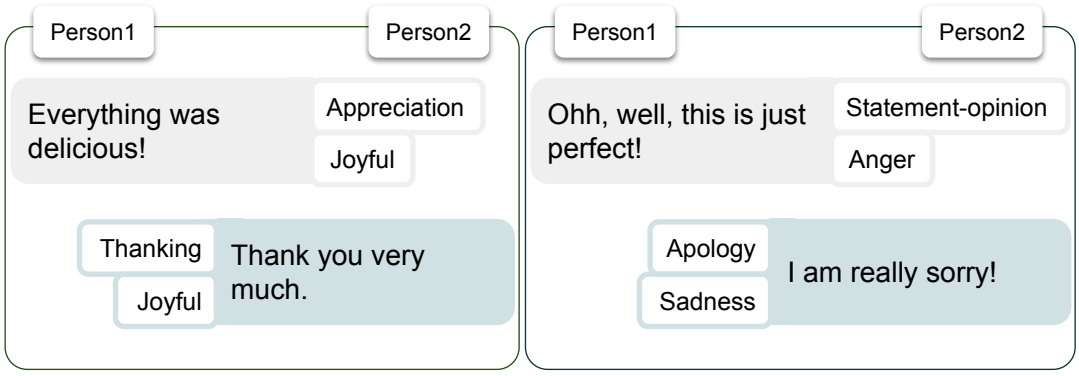


Figure 6.1: Example of a contextual dialogue with dialogue acts (upper box aside utterance) and emotion expressions (lower box aside utterance).

extracts the final dialogue act label. We annotated two accessible multi-modal emotion datasets: IEMOCAP and MELD. We analyzed the co-occurrence of emotion and dialogue act labels and discovered specific relations. For example, *Accept/Agree* dialogue acts often occur with the *Joy* emotion, *Apology* with *Sadness*, and *Thanking* with *Joy*. We make the Emotional Dialogue Act (EDA) corpora publicly available to the research community for further study and analysis.

With the growing demand for human-computer/robot interaction systems, detecting the user's emotional state can primarily benefit a conversational agent to respond at an appropriate affective level. Emotion recognition in conversations has proven valuable for various applications such as response recommendation or generation, emotion-based text-to-speech, and personalization. Human emotional states can be expressed verbally and non-verbally (Ekman et al., 1987; Osgood et al., 1975), and however, while building an interactive dialogue system, the interface needs dialogue acts to understand user input utterance (López-Cózar et al., 2010). A typical dialogue system consists of a language understanding module that requires determining the meaning and intention in the input utterances (Berg, 2015; Ultes et al., 2017). Also, in conversational discourse analysis, dialogue acts are the main linguistic features to consider (Bothe et al., 2018b). The dialogue act provides an intention and performative function in the utterance of a dialogue. For example, it can distinguish different intentions such as *Question*, *Answer*, *Request*, and *Agree/Reject* and performative functions such as *Acknowledgement*, *Conversational-opening or -closing*, *Thanking* and *Apology*.

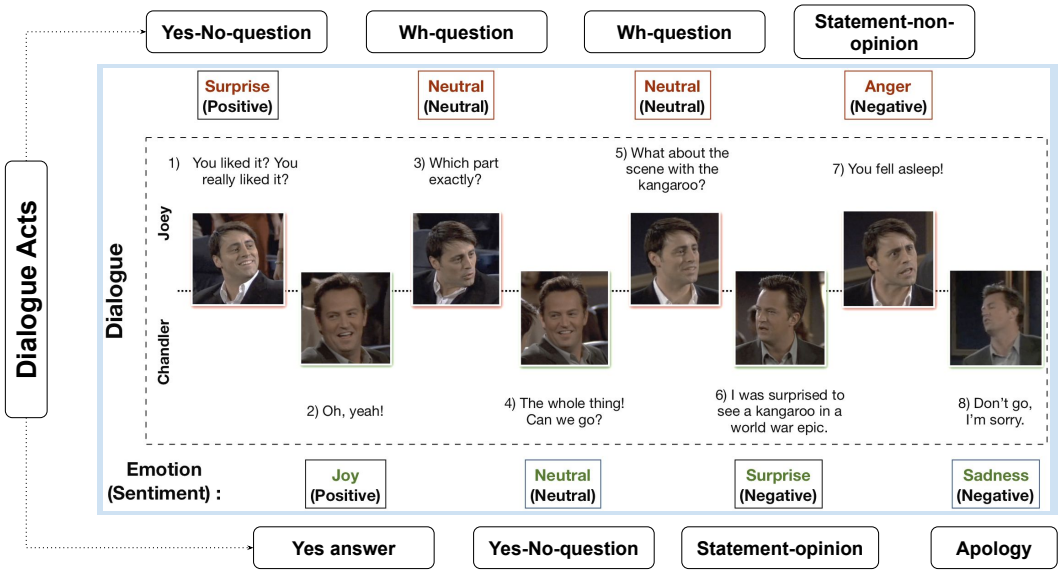


Figure 6.2: Emotional Dialogue Acts: Example of a dialogue from MELD representing emotions and sentiment (rectangular boxes), in our work, we add dialogue acts (rounded boxes) to the original image from Poria et al. (2019).

The dialogue act information, together with emotional states, can be beneficial for a spoken dialogue system to produce natural affective interaction (Ihasz and Kryssanov, 2018).

The research in emotion recognition is proliferating, and many datasets are available, such as text-based, speech- or vision-level, and multi-modal conversational emotion data. Emotion expression recognition is a challenging task, and hence multimodality is crucial (Ekman et al., 1987). However, a few conversational multi-modal emotion datasets are available, for example, IEMOCAP (Busso et al., 2008), SEMAINE (McKeown et al., 2012), MELD (Poria et al., 2019). They are multi-modal dyadic conversational datasets containing audio-visual and conversational transcripts. Every utterance transcript in these datasets is labelled with an emotion label.

In this chapter, we explore an automated neural ensemble annotation process for dialogue act labelling. Several neural models are trained with the SwDA corpus (Godfrey et al., 1992; Jurafsky et al., 1997) and used for inferring dialogue acts on the emotion datasets. We ensemble the outputs of the five models by checking majority occurrences (most of the model outputs the same label)

and ranking confidence values of the models. We have annotated two potential multi-modal conversation datasets for emotion recognition: IEMOCAP (Interactive Emotional dyadic MOtion CAPture database) and MELD (Multimodal EmotionLines Dataset). Figure 6.2, shows an example of the dialogue act tags with emotion and sentiment labels from the MELD dataset. We confirmed the reliability of annotations with several inter-annotator metrics. We analyzed the co-occurrences of the dialogue act and emotion labels and discovered an interesting relationship between them; individual dialogue acts of the utterances show significant and useful association with respective emotional states. For example, *Accept/Agree* dialogue act often occurs with the *Joy* emotion while *Reject* with *Anger*, *Acknowledgements* with *Surprise*, *Thanking* with *Joy*, and *Apology* with *Sadness*. The detailed analysis of the emotional dialogue acts (EDAs) are reported in this chapter, and annotated datasets are available at the SECURE EU Project website[1].

6.2 Annotation of Emotional Dialogue Acts (EDA)

6.2.1 Data for Conversational Emotion Analysis

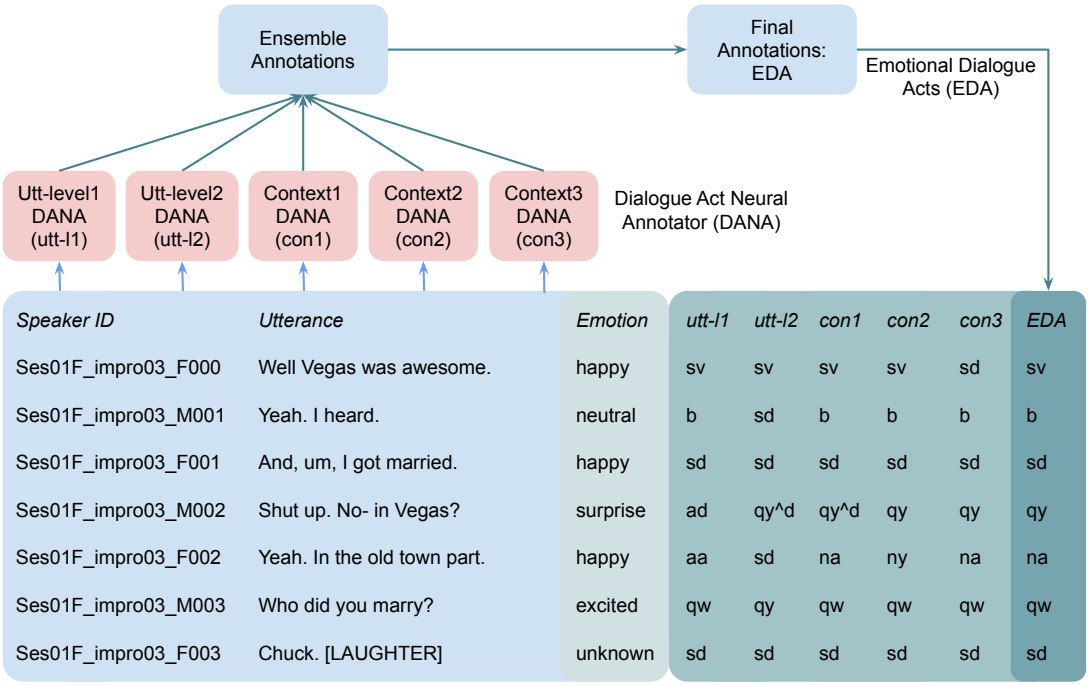
There are two primary emotion taxonomies: (1) discrete emotion categories (DEC) and (2) fined-grained dimensional basis of emotion states (DBE). The DECs are Joy, Sadness, Fear, Surprise, Disgust, Anger and Neutral; identified by Ekman et al. (Ekman et al., 1987). The DBE of the emotion is usually elicited from two or three dimensions (Osgood et al., 1975; Russell and Mehrabian, 1977; Cowie and Cornelius, 2003). A two-dimensional model is commonly derived with Valence and Arousal (also called activation), and the three-dimensional model contains Dominance as a third dimension. IEMOCAP is annotated with all DECs and two additional emotion classes, Frustration and Excited. IEMOCAP is also annotated with three DBE, that includes Valance, Arousal and Dominance (Busso et al., 2008). MELD (Poria et al., 2019), which is an evolved version of the EmotionLines dataset developed by (Chen et al., 2018a), is annotated with exactly 7 DECs and sentiment labels (positive, negative and neutral).

[1] `https://secure-robots.eu/fellows/bothe/EDAs/`, IEMOCAP is available only with speaker IDs, for full data visit `https://sail.usc.edu/iemocap/`

6.2.2 DA Tagset and SwDA Corpus

As discussed in Chapter 4, there have been many taxonomies for dialogue acts: speech acts (Austin, 1962) refer to the utterance, not only to present information but also to the action performed by an utterance. Speech acts were later modified into five classes (Assertive, Directive, Commissive, Expressive, Declarative) (Searle, 1979). Many such standard taxonomies and schemes are used to annotate conversational data, and most of them follow the discourse compositionality. These schemes have proven their importance for conversational discourse analysis (Skantze, 2007). During the increased development of dialogue systems and discourse analysis, the standard taxonomy was introduced in recent decades, the DAMSL tag set being one of them. As discussed in chapter 2, each DA has a forward-looking function (such as Statement, Info-request, Thanking) and a backwards-looking function (such as Accept, Reject, Answer) (Allen and Core, 1997). The DAMSL annotation includes not only the utterance-level but also segmented-utterance labelling.

However, in the emotion datasets, the utterances are not segmented, as we can notice from Figure 6.2, first and fourth utterances are not segmented as two separate ones. The fourth utterance could be segmented to have two different dialogue act labels, for example, Statement (*sd*) and Yes-No Question (*qy*). That could provide very fine-grained DA classes and follows the concept of discourse compositionality. DAMSL scheme distinguishes Wh-question *(qw)*, Yes-No question *(qy)*, Open-ended question *(qo)*, and Or-question *(qr)* dialogue act classes, not just because these questions are syntactically distinct, but also because they have different forward functions (Jurafsky, 1997). For example, yes-no question (*qy*) is more likely to get a *"yes"* answer than a Wh-question *(qw)*. It also gives an intuition that the answers follow the syntactic formulation of the question, providing a context. For example, *qy* is used for a question that, from a discourse perspective, expects a Yes- *(ny)* or No- *(nn)*Answer dialogue act. We have investigated the annotation methods (in Section 6.2) and the neural models are trained with the SwDA corpus (Godfrey et al., 1992; Jurafsky et al., 1997). SwDA corpus is annotated with the DAMSL tag set, and it has been used for reporting and bench-marking state-of-the-art results in the dialogue act recognition task (Stolcke et al., 2000; Kalchbrenner and Blunsom, 2013b; Bothe et al., 2018d) which makes it ideal for this use case of the ensemble of neural annotators. The details



Speaker ID	*Utterance*	*Emotion*	*utt-l1*	*utt-l2*	*con1*	*con2*	*con3*	*EDA*
Ses01F_impro03_F000	Well Vegas was awesome.	happy	sv	sv	sv	sv	sd	sv
Ses01F_impro03_M001	Yeah. I heard.	neutral	b	sd	b	b	b	b
Ses01F_impro03_F001	And, um, I got married.	happy	sd	sd	sd	sd	sd	sd
Ses01F_impro03_M002	Shut up. No- in Vegas?	surprise	ad	qy^d	qy^d	qy	qy	qy
Ses01F_impro03_F002	Yeah. In the old town part.	happy	aa	sd	na	ny	na	na
Ses01F_impro03_M003	Who did you marry?	excited	qw	qy	qw	qw	qw	qw
Ses01F_impro03_F003	Chuck. [LAUGHTER]	unknown	sd	sd	sd	sd	sd	sd

Figure 6.3: Setup of the annotation process of the EDAs, above example utterances (with speaker identity) and emotion labels are from IEMOCAP database.

about this dataset is provided in Chapter 4 and in Appendix B.1.

6.2.3 Neural Annotators

We adopted the neural architectures based on (Bothe et al., 2018c) where the two primary variants are non-context model and context model. The non-context model classifies at an utterance-level, whereas the context model uses the current utterance along with a few preceding utterances for the classification. From conversational analysis using dialogue acts in (Bothe et al., 2018b), we learned that the preceding two utterances contribute significantly to recognizing the dialogue act of the current utterance. Hence, we adapt this setting for the context model and create a pool of annotators using recurrent neural networks (RNNs). RNNs can model the contextual information in the sequence of words of an utterance, and the sequence of utterances of a dialogue. Each word of the utterance is represented with a word embedding vector of dimension 1024 using the word embedding vectors from the pre-trained ELMo (Embeddings from Language Models)

embeddings[2] (Peters et al., 2018). We create a pool of five neural annotators, as shown in Figure 6.3. Our online tool called Discourse-Wizard[3] is available to demonstrate automated dialogue act labelling. We use the same neural architectures in the backend in this tool, and the entire process is encapsulated with ELMo embeddings as a REST API. The annotators are shown in Figure 6.4 that represents following models:

Utt-level 1 Dialogue Act Neural Annotator (DANA) is an utterance-level classifier that uses word embeddings (w) as an input to the RNN layer with attention mechanism (a) and computes the probability of dialogue acts (da) using *softmax* function (see in Figure 6.4, dotted line *utt-l1*). This model achieved 75.13% accuracy on the SwDA corpus test set.

Context 1 DANA is a context model that uses two preceding utterances while recognizing the dialogue act of the current utterance (see context model with *con1* line in Figure 6.4). It uses a hierarchical RNN architecture with the first RNN layer to encode the utterance from word embeddings (w) and the second RNN layer is provided with these encoded utterances (u), current and two preceding ones, followed by the attention mechanism (a), where $\sum_{n=0}^{n} a_{t-n} = 1$. Finally, the *softmax* function is used to compute the probability distribution of the dialogue acts (da). This model achieved 77.55% accuracy on the SwDA corpus test set.

Utt-level 2 DANA is another utterance-level classifier which takes an average of the word embeddings in the input utterance and uses a feedforward neural network hidden layer (see the *utt-l2* line in Figure 6.4, where mean (*avg.*) passed directly to the *softmax* function). Similar to the previous model, it computes the probabilities of dialogue acts using the *softmax* function. This model achieved 72.59% accuracy on the test set of the SwDA corpus.

Context 2 DANA is another context model that uses three utterances similar to the Context 1 DANA model, but the utterances are composed as the mean of the word embeddings over each utterance, similar to the Utt-level 2 model (*avg.* passed to context model in Figure 6.4 with *con2* line). Hence, the Context 2 DANA model is composed of one RNN layer with three input vectors, finally topped with the *softmax* function for computing the probability distribution of the dialogue acts. This model achieved 75.97% accuracy on the SwDA corpus test set.

[2] `https://allennlp.org/elmo`

[3] `https://secure-robots.eu/fellows/bothe/discourse-wizard-demo/`

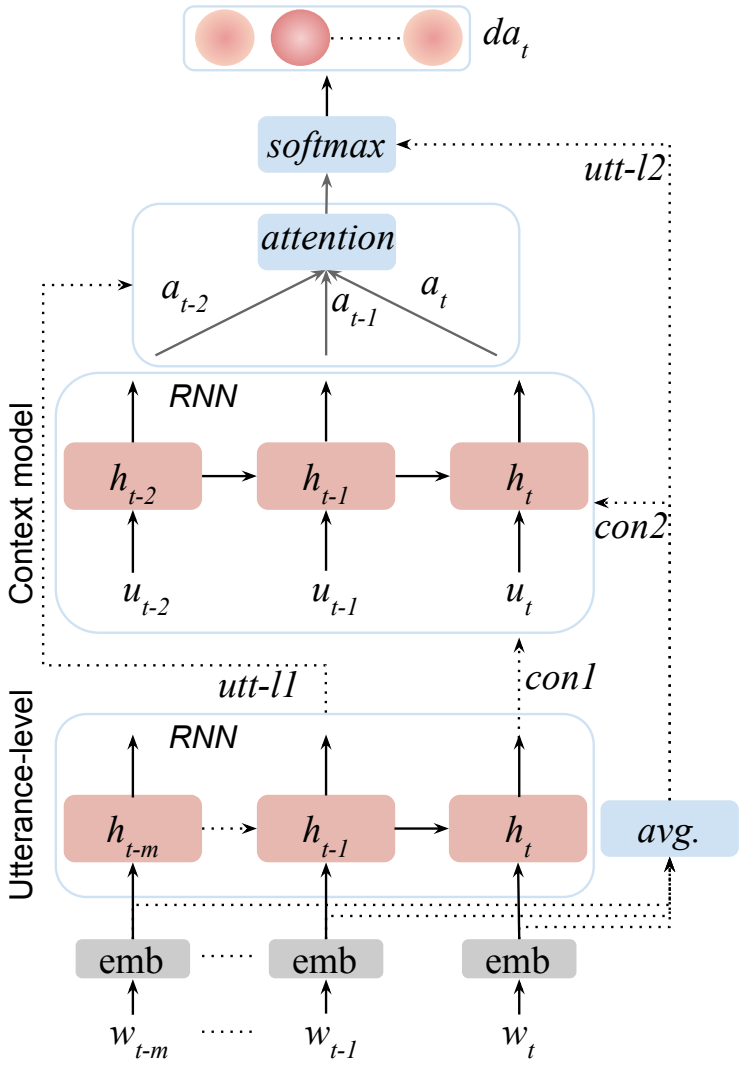


Figure 6.4: Recurrent neural network mechanism with attention mechanism depicting all the architectures of the utterance-level and context-based models.

Context 3 DANA is third context model that uses three utterances similar to the previous context models, but the utterance representations combine both features from the Context 1 and Context 2 models (*con1* and *con2* together in Figure 6.4). Hence, the Context 3 DANA model combines features of almost all the previous four models to provide the recognition of the dialogue acts. This model achieves 75.91% accuracy on the SwDA corpus test set.

Stats	AllMatch	ConModel	ConfMatch	NoMacth
IEMOCAP	43.73%	46.66%	3.01%	6.60%
MELD	37.07%	47.20%	4.58%	11.15%

Table 6.1: Annotations Statistics of EDAs - AllMatch: All Models Absolute Match, ConModel: Context-based Models Absolute Match (matched all context models or at least two context models matched with one non-context model), ConfMatch: Based-on Confidence Ranking, and NoMacth: No Match (these are labeled as 'xx': determined in EDAs).

6.2.4 Ensemble of Neural Annotators

First preference is given to the labels that are perfectly matching the predictions of all the neural annotators. Table 6.1 shows that both datasets have about 40% of exactly matching labels over all the models. Then priority is given to the context-based models to check if the label in predictions of all context models is matching perfectly. In case two out of three context models match correctly, and if the same label is also produced by at least one of the non-context models, we allow these labels to rely on these at least two context models. As a result, about 47% of the labels are taken based on the context models.

When we see that none of the context models is producing the same labels, then we rank the labels with their respective confidence values produced as the probability distribution using the $softmax$ function. The labels are sorted in descending order according to the confidence values. Then we check if the first three (case when one context model and both non-context models produce the same label) or at least two labels are matching, then we allow to pick that one. There are about 3% in IEMOCAP and 5% in MELD.

Finally, when none of the above conditions is fulfilled, we leave the label with an unknown category. This unknown category of the dialogue act is labelled with *'xx'* in the final annotations, and they are about 7% in IEMOCAP and 11% in MELD. The statistics[4] of the EDAs is reported in Table 6.3 for both datasets. The total utterances in the annotated MELD corpus include the training, validation

[4]The updated statistics and datasets are available at: `https://github.com/bothe/EDAs`

Metrics	α	k	SCC
IEMOCAP	0.553	0.556	0.636
MELD	0.494	0.502	0.585

Table 6.2: Annotations Metrics of EDAs - α: Krippendorff's Alpha coefficient, k: Fleiss' Kappa score, and SCCM: Spearman Correlation between first two Context-based Models.

and test sets[5].

6.2.5 Reliability of Ensemble Neural Annotators

The pool of neural annotators provides an acceptable range of annotations, and we checked the reliability with the following metrics (McHugh, 2012). Krippendorff's Alpha (α) is a reliability coefficient which is often used in emotion annotation (Wood et al., 2018). It is created to measure the agreement among annotators, and raters. We apply it on the five neural annotators at the nominal level of measurement of dialogue act categories. α is computed as follows:

$$\alpha = 1 - \frac{D_o}{D_e} \tag{6.1}$$

where D_o is the observed disagreement and D_e is the disagreement that is expected by chance. $\alpha = 1$ means all annotators produce the same label, while $\alpha = 0$ would mean none agreed on any label. As we can see in Table 6.2, both the datasets, IEMOCAP and MELD, produce significant inter-neural-annotator agreement, 0.553 and 0.494, respectively.

A prevalent inter-annotator metric is Fleiss' Kappa score, also reported in Table 6.2, which determines consistency in the ratings. The kappa score k can be calculated as,

$$k = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e} \tag{6.2}$$

where the numerator $\bar{P} - \bar{P}_e$ provides the degree of actually achieved agreement over the denominator $1 - \bar{P}_e$ that elicits the degree of agreement that is attainable above possibility. Hence, $k = 1$ if the raters agree completely, and $k = 0$

[5] `https://affective-meld.github.io/`

when none reach any agreement. We got 0.556 and 0.502 for IEOMOCAP and MELD, respectively, with our five neural annotators. It indicated that the annotators are labelling the dialogue acts reliably and consistently. We also report the Spearman's correlation between context-based models (Context1 and Context2), and it shows a strong correlation between them (see in Table 6.2). While using the labels, we checked the absolute match between all context-based models and hence their strong correlation indicates the robustness of the neural annotators.

6.3 EDAs Analysis

We can see emotional dialogue act co-occurrences with respect to emotion labels in Figure 6.5 for both datasets. There are sets of three bars per dialogue act in the figure, the first and second bar represents emotion labels of IEMOCAP (IE) and MELD (ME), respectively, and the third bar is for MELD sentiment (MS) labels. MELD emotion and sentiment statistics are compelling as they are strongly correlated to each other. The bars contain the normalized number of utterances for emotion labels regarding the total number of utterances for that particular dialogue act category. The statements without-opinion (*sd*) and with-opinion (*sv*) contain utterances with almost all the emotion labels and many neutral emotion utterances are spanning over all the dialogue acts.

On the other hand, the quotation ($\hat{}q$) dialogue act labelled utterances are mostly used with 'Anger' and 'Frustration' (in case of IEMOCAP), however, some utterances with 'Joy' or 'Sadness' as well (see examples in Table 6.4). Action Directive (*ad*) dialogue act utterances, which are usually commands or orders, frequently occur with 'Anger' or 'Frustration' although many with 'Happy' emotion in case of MELD. Acknowledgements (*b*) are mostly with positive or neutral sentiment, however, Appreciation (*ba*) and Rhetorical (*bh*) backchannels often occur with a greater number in 'Surprise', 'Joy' and/or with 'Excited' (in case of IEMOCAP). Questions (*qh, qw, qy and qy*$\hat{}$*d*) are mostly asked with emotions 'Surprise', 'Excited', 'Frustration' or 'Disgust' (in case of MELD), and many are neutral. No-answers (*nn*) are mostly 'Sad' or 'Frustrated' as compared to Yes-answer (*ny*). Forward-functions such as Apology (*fa*) are mostly with 'Sadness' whereas Thanking (*ft*) and Conventional-closing or -opening (*fc* or *fp*) are usually with 'Joy' or 'Excited'. We also noticed that both datasets exhibit a very similar relationship between dialogue act and emotion labels. It is essential to no-

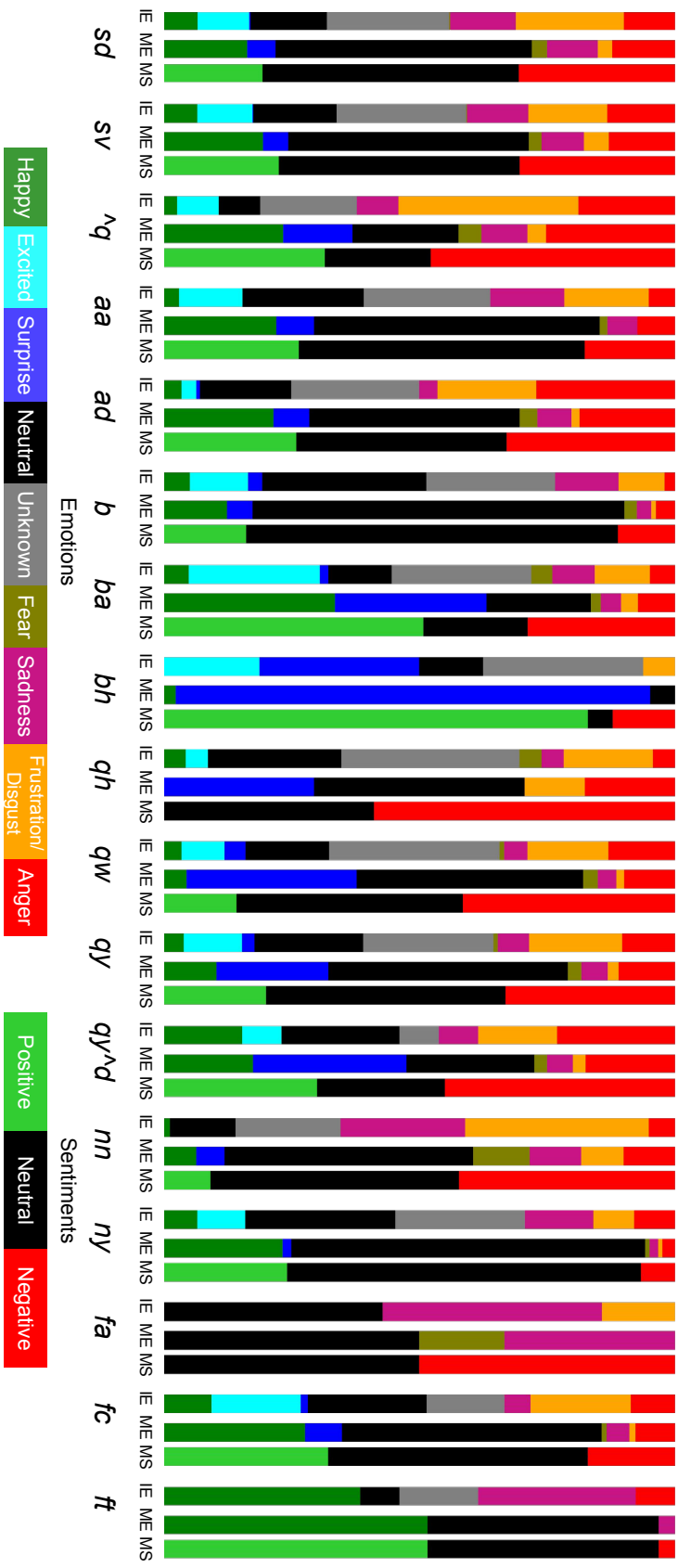


Figure 6.5: EDAs: Visualizing co-occurrence of utterances with respect to emotional states in the particular dialogue acts (only major and significant are shown here). IE: IEMOCAP Emotion, ME: MELD Emotion and MS: MELD Sentiment.

DA	Dialogue Act	IEMOCAP	MELD
sd	Statement-non-opinion	43.97	41.63
sv	Statement-opinion	19.93	09.34
qy	Yes-No-Question	10.3	12.39
qw	Wh-Question	7.26	6.08
b	Acknowledge (Backchannel)	2.89	2.35
ad	Action-directive	1.39	2.31
fc	Conventional-closing	1.37	3.76
ba	Appreciation or Assessment	1.21	3.72
aa	Agree or Accept	0.97	0.50
nn	No-Answer	0.78	0.80
ny	Yes-Answer	0.75	0.88
br	Signal-non-understanding	0.47	1.13
^q	Quotation	0.37	0.81
na	Affirmative non-yes answers	0.25	0.34
qh	Rhetorical-Question	0.23	0.12
bh	Rhetorical Backchannel	0.16	0.30
ft	Thanking	0.13	0.23
qy^d	Declarative Yes-No-Question	0.13	0.29
bf	Reformulate	0.12	0.19
fp	Conventional-opening	0.12	1.19
fa	Apology	0.07	0.04
fo	Other Forward Function	0.02	0.05
	Total number of utterances	10039	13708

Table 6.3: Number of utterances per DA in respective datasets. All values are in percentages (%) of the total number of utterances.

EDAs	Utterances	Emotion
Quotation (^*q*)	Not after this!	anger
	Ross, I am a human doodle!!	anger
	No, you can't let this stop!	sadness
	Oh hey! You got my parent's gift!	joy
Action-Directive (*ad*)	And stop using my name!	anger
	Oh, let's not tell this story.	sadness
	Check it out, he's winning!	surprise
	Yep! Grab a plate.	joy
Backchannel (*b*)	Oh yeah, sure.	neutral
Appreciation *b* (*ba*)	Great.	joy
Rhetorical *b* (*bh*)	Oh really?!	surprise
Rhetorical Question (*qh*)	Oh, why is it unfair?	surprise
Wh-Question (*qw*)	What are you doing?	surprise
	How are you?	neutral
Yes-No Question (*qy*)	Did you just make that up?	surprise
Declarative *qy* (*qy*^*d*)	Can't you figure that out?	anger
No-Answer (*nn*)	No!	disgust
Yes-Answer (*ny*)	Yeah!	joy

Table 6.4: Examples of EDAs with annotation from the MELD dataset. Emotion and sentiment labels are given in the dataset, while our ensemble of models annotates EDAs.

tice that the dialogue act annotation is based on the given transcripts; however, the emotional expressions are better perceived with audio or video (Busso et al., 2008).

We report some examples where we mark the utterances with a determined label (*xx*) given in Table 6.5. They are skipped from the final annotation because of not fulfilling the conditions of the ensemble model explained in Section 6.2.4

EDAs	Utterances	Emotion
Determined EDAs (*xx*)		
1. (P-DA *b*) *b*, *b*, *ba*, *fc*, *b*	Yeah, sure!	neutral
2. (P-DA *sd*) *sv*, *aa*, *bf*, *sv*, *nn*	No way!	surprise
3. (P-DA *qy*) *aa*, *aa*, *ng*, *ny*, *nn*	Um-mm, yeah right!	surprise
4. (P-DA *qy*) *aa*, *ar*, *^q*, *^h*, *nn*	Oh no-no-no, give me some specifics.	anger
5. (P-DA *fc*) *fc*, *sd*, *fc*, *sd*, *fp*	I'm so sorry!	sadness

Table 6.5: Examples of determined EDAs with annotation from the MELD dataset. Emotion/sentiment labels are given in the dataset, while EDAs are by our ensemble of models. P-DA: previous utterance dialogue act.

It is also interesting to see the previous utterance dialogue acts (P-DA) of those skipped utterances, and the labels of the neural annotators as given in Figure 6.3 (utt-l1, utt-l2, con1, con2, con3). In the first example, the previous utterance is *b*, and three DANA models produced labels of the current utterance as *b*, but it is skipped because the confidence values could not bring it as a final label. The second utterance can be challenging even for humans to perceive with any of the dialogue acts. However, the third and fourth utterances are followed by a Yes-No question (*qy*), and hence, we can see in the third example, the context models tried to at least perceive it as an answer dialogue acts (*ng, ny, nn*). The last utterance, "I'm so sorry!", has reasonable disagreement by all the five annotators. Similar apology phrases are mostly found with 'Sadness' emotion labels, and the correct dialogue act is Apology (*fa*). However, these utterances are placed either in the *sd* or in *ba* dialogue act category. We believe that those labels of the utterances can be corrected with minimal efforts with human annotator's help.

6.4 Summary

In this chapter, we presented a method to extend conversational multi-modal emotion datasets with dialogue act labels. We successfully show this on two well-

established emotion datasets: IEMOCAP and MELD, which we labelled with dialogue acts and made publicly available for further study and research. As a first insight, we found that many dialogue acts and emotion labels follow certain relational features. These relations can be useful to learn about the emotional behaviours with dialogue acts to build a natural dialogue system and perform deeper conversational analysis. The conversational agent might benefit in generating an appropriate affective response when considering both emotional states and dialogue acts of the utterances.

In future work, we foresee the human in the loop for the annotation process along with a pool of automated neural annotators. Robust annotations can be achieved with minimal human effort and supervision, for example, observing and correcting the final labels produced by ensemble output labels from the neural annotators. The human-annotator might also help to achieve segmented-utterance labelling of the dialogue acts. We also plan to use these datasets for conversational analysis to infer interactive behaviours of the emotional states with respect to the dialogue acts. In the next experiment, we use dialogue acts to build a dialogue system for a social robot, where we find this study and dataset very helpful. For example, we can extend our robotic conversational system to consider emotion as an added linguistic feature alongside politeness to produce natural interaction.

Chapter 7

Dialogue-based Navigation driven by Politeness for HRI

Service robots need to show appropriate social behaviour in order to be deployed in social environments such as healthcare, education, and retail. Some of the main capabilities that robots should have are navigation and conversational skills. If the person is impatient, that can act as a cue for the robot to navigate faster and vice versa. Linguistic features that indicate politeness can provide social cues about a person's patient and impatient behaviour. The novelty presented in this experiment is to incorporate the politeness feature in a robotic dialogue system for a dynamic navigation speed. Understanding the politeness cues in users' utterance can also be used to modulate the robot behaviour and responses accordingly. Therefore, we developed a dialogue system to navigate the humanoid robot in an indoor environment, which produces different robot behaviours and responses based on the users' intention and degree of politeness. We tested our system with the Pepper humanoid robot that adapts to the changes in users behaviour at the Innovation Department Lab of SoftBank Robotics Europe.

7.1 Introduction

In this experiment, we develop a dialogue system with multivariate behavioural adaptation based on the socio-linguistic aspects such as politeness. It is another preliminary step towards understanding the safety concepts for safe human-robot interaction (HRI). Politeness strategies ensure smooth communication and harmonious interpersonal relationship in non-hostile social communication (Kamlasi,

2017). Politeness reflects the perception of the users towards their patience during the interaction. Hence, perceiving politeness of the user becomes a crucial process in HRI. In addition to other factors, such as robot appearance, robot behaviour is a crucial aspect of their acceptance in society. This work is a primary step towards developing a dialogue system for safe and pro-active HRI where the robot takes advantage of linguistic politeness comprehension to adapt social behaviours. The socio-linguistic factors like politeness also play an essential role in knowing whether social interaction goes appropriately or poorly. Hence, politeness cues are intimately related to the dynamics of behavior and interaction (Brown and Levinson, 1987; Danescu-Niculescu-Mizil et al., 2013; Holmes and Stubbe, 2015; Srinivasan and Takayama, 2016). It is useful for adapting to the dynamic tension that occurs as the user tries to maintain a sufficient degree of politeness while interacting with the robot (Rogers and Lee-Wong, 2003). For example, sentence-initial *you* or an action directive verb can be impolite *"You need to show..."* or *"Show me the..."*, whereas sentence-medial *you* or sentence-initial *could* or *would* often indicates polite interaction like in the sentences *"Could you show me..."* or *"Would you take me to..."*.

Multivariate adaptive and affective dialogue systems based on linguistic features have been subject to previous research (Fong et al., 2003a; Adam et al., 2016; Shi and Yu, 2018). The effect of politeness on the conversation is prominent, and it has been researched in the socio-linguistic community (Rogers and Lee-Wong, 2003; Danescu-Niculescu-Mizil et al., 2013; Holmes and Stubbe, 2015). The effect of such features on HRI has been a subject of study with various aspects: polite versus impolite robot playing a game (Castro-González et al., 2016), in determining social robot acceptance with multi-cultural background people (Salem et al., 2014), making robots sociable and supporting to achieve safe HRI (Fong et al., 2003a). Hence, a robot that can recognize the user's intention during interaction should also adapt to the human's linguistic behavioural changes. For example, different socio-linguistic features, such as politeness, emotion, and sentiment, represent users' social and behavioural dynamic interactions.

For this experimental HRI scenario, we portray the politeness concept as a linguistic cue. If the users' utterance is impolite, then the user might be impatient, that means the user is in a hurry or urgency as one uses short phrases in the utterances in such a situation. On the other hand, if the user is polite, the behaviour can be another way around. In such cases, the robot needs to change

its behaviour or even alter the actions and speed to maintain social and safe adjustments during the interaction. The modulated behaviours can be programmed to intrigue the user to maintain the engagement with the robot. We develop a modular dialogue system (DS) that can process such features and make the robot to adapt accordingly. The natural language understanding module of the DS uses recurrent neural networks (RNNs) (Ultes et al., 2017; Yang et al., 2017) and the Snips library (Coucke et al., 2018) for extracting the intentions and structured information from the user input utterances. The politeness detection is learned from the corpus using RNNs and fine-tuned for the domain-specific data. The navigation of the Pepper humanoid robot is achieved by using the NAOqi framework. The robot behaviour and responses are driven by dialogue flow module using the intention and politeness detection in the input utterances. This experiment's main contribution towards bridging the gap between socio-linguistic research and HRI community is in understanding the use of the socio-linguistic feature for developing the dialogue-based navigation system that incorporates politeness as a primary driving social cue. As a future work direction, this experiment provides a robotic behavioural model based on literature that can be assessed with extended experimentation. To the best of our knowledge, our system is the first dialogue-based navigation system that incorporates politeness as a primary social cue to drive the robot behaviour and responses.

7.2 Approach: Proposed HRI Dialogue System

We propose a dialogue system which takes into account the degree of politeness as a factor that affects the conversational flow and the robots' behaviour. Usually, a typical dialogue system considers only the intention and semantic information extracted from the utterance such as slot-value pairs (discussed in the next section) (Ultes et al., 2017). We use politeness detection module as an additional driving feature in this particular experiment, and it could be scaled to extend to use other social cues such as emotion. In the proposed model, the dialogue system can process various socio-linguistic features to perform inference on the input utterance. The overall architecture is shown in Figure 7.1. As mentioned, the proposed system is customizable to any extent as the robot is controlled using a modular client-server architecture. The state and motion managers are wrapped into an application programming interface (API) as a server (Grinberg, 2018) and

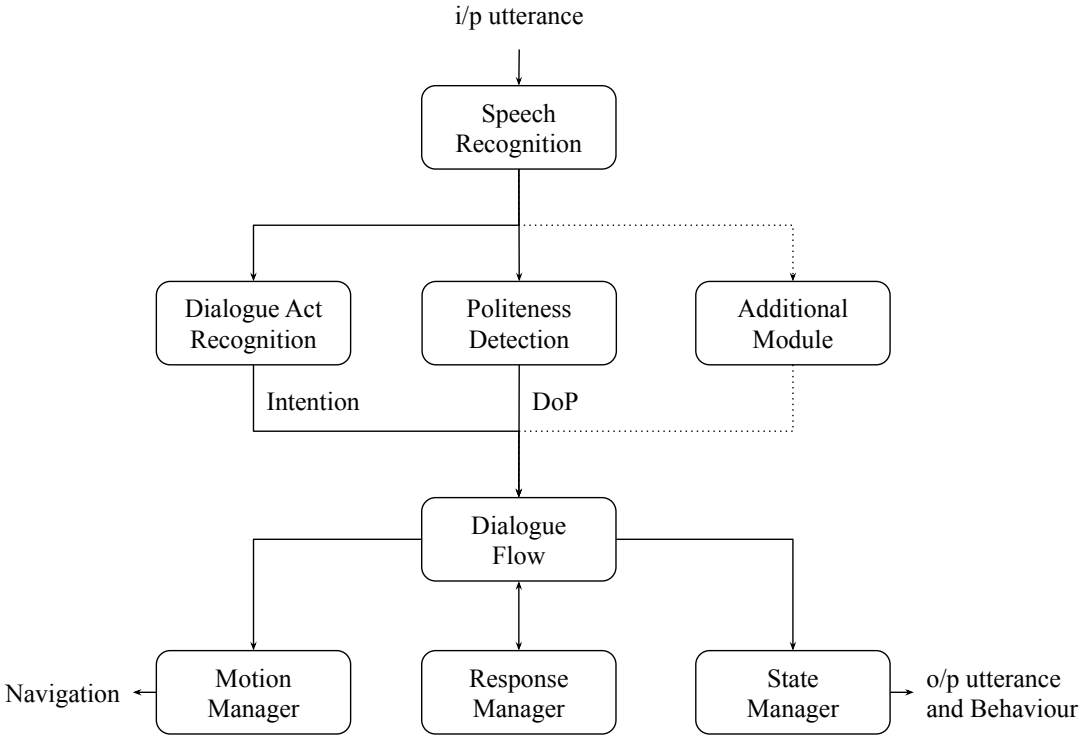


Figure 7.1: The overall architecture of the dialogue system. Degree of Politeness (DoP) is the primary driven of the conversational behaviour.

communicated via the client dialogue flow module. The dialogue system can also be accessed if the robot is not connected to the server, which is useful for human-computer interaction scenario, making the dialogue system similar to the one explained in Section 2.3. However, it is an advance version of the typical dialogue system as we incorporate socio-linguistic cues in the proposed architecture.

The dialogue system takes the input utterance of the human user through a speech recognition system. The speech recognition used is embedded on the Pepper robot platform with an independent server API. The transcribed text is then processed by the dialogue act recognition and politeness detection modules, and additional modules. Dialogue Flow (DF) module takes the output from these modules to interpret the response from the Response Manager. The DF module also sends its interpretation and the fetched response to the Motion Manager and State Manager to variate the robot navigation speed and behaviours accordingly. Motion Manager contains the information of the planned map and a programmed switch for the speed based on the degree of politeness (discussed in later sections).

State Manager keeps track of overall behavioural state and adjusts parameters of the robot such as speech pitch, head orientation angle, and eyes colour, and also, utters the output response.

7.3 Dialogue System driven by Politeness

The conversational part of the dialogue system (DS) can be used independently of the robot. The central part of the DS is the language understanding module which consists of the Dialogue Act Recognition, Politeness Detection, and additional modules. Another central part of the proposed DS is the Dialogue Flow module which takes care of the flow of utterances and sending commands to the robot. Finally, the response manager is responsible for interpreting an appropriate response given the intention and the degree of politeness.

7.3.1 NLU: Intention and Politeness Detection

The input speech from a user is converted into text using the embedded speech recognition module from the Pepper robot (accessed via the NAOqi framework). The natural language understanding (NLU) module takes the converted transcript of the input utterance and processes it with the dialogue act recognition and politeness detection modules. DA recognition module detects intention and semantic information in the utterance, whereas politeness detection module infers politeness polarity of that utterance.

DA Recognition Module

The dialogue act (DA) recognition is a crucial process in any typical dialogue system. Its task is to decode the natural language input utterance and extract the symbolic representation, such as dialogue acts and slot-value pairs. For example, the utterance *"Could you please show me the retail department?"* can be decoded as $\{intention : TakeToPlace, department : retail\}$ where *intention* represents the dialogue act, *department* is a slot and *retail* being its value. It is also called the user dialogue act because the users' input utterances determine it. We created a dataset for the given scenario to be able to drive the conversation (some examples are given in Table 7.1). The following methods are used in conjunction for robust

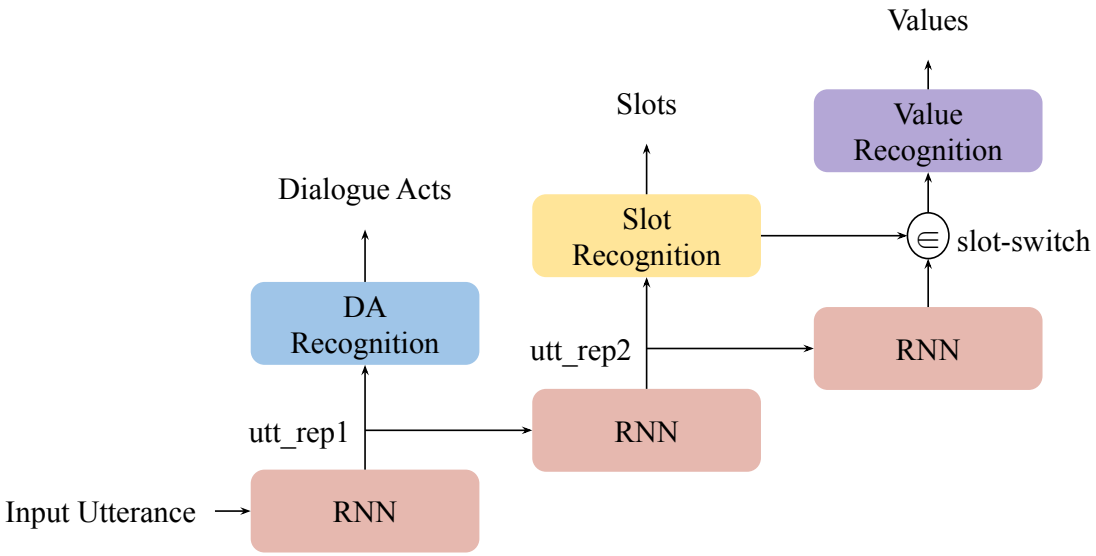


Figure 7.2: Dialogue acts and slot-value pairs recognition using RNNs.

dialogue act recognition by validating one another based on heuristics of their confidence values.

(1) Dialogue Act Recognition using Hierarchical RNNs: We discussed the basic hierarchical RNN (HiRNN) architecture and its brief operation in Section 3.2.3. As mentioned there, the hidden unit representations of each RNN layer in the HiRNN architecture can be used solely and could also be carried forward for the other task learning. A customized architecture is shown in Figure 7.2, where RNNs are used in hierarchical fashion to learn the dialogue acts and slot-value pairs (Yang et al., 2017; Bothe et al., 2018c; Kumar et al., 2018). As we can see in the architecture, the first layer of the RNN is used to classify the dialogue acts, and the utterance representation *utt_rep1* is carried forward to the next RNN layer. The next RNN layer uses *utt_rep1* representation to recognize slots, producing a new hidden utterance representation *utt_rep2*. The values of the slots are learned with the next RNN layer using *utt_rep2* representation. The slot-switch ($\in$) is used for classifying the values learned in this layer to the slot detected in the past layer (see the output in Figure 7.5 for better understanding). We train the model to the customized data and use them for inference.

Dialogue acts	Examples	Slots	Values
Greeting	*Hello.*	no_slot	no_value
	Hi, how are you?		
Thanking	*Thank you.*	no_slot	no_value
	Thank you very much.		
TakeToPlace	*Could you show me the education department?*	room	retail
	Take me to the retail section.		education
	Can you take me to tourism department?		tourism
MoveRobot	*Please go ahead.*	direction	forward
	Could you move ahead?		backward
	Go back please.		
TurnRobot	*Can you turn right?*		right
	Could you turn left.		left
Accept	*Yes, I would like to visit.*	no_slot	no_value
AbortRobot	*stop, wait, be careful...*	no_slot	no_value

Table 7.1: Examples of dialogue act and slot-value pairs

(2) Snips Natural Language Understanding (NLU) Engine: Snips NLU Engine[1] is an open-source Python library that uses two approaches: a deterministic parser and a probabilistic parser (Coucke et al., 2018). The deterministic parser is a pattern matching mechanism which uses regular expressions to parse the input utterance. The probabilistic parser uses a logistic regression algorithm

[1] `https://snips-nlu.readthedocs.io`

DoP	Class	Utterance
1	polite	Could you please show me the education department?
0	neutral	Can you show me the education department?
-1	impolite	Show me the education department.

Table 7.2: Examples of utterances in different Politeness classes.

for intent classification and conditional random fields (CRFs) technique for the slot filling task. For the given input utterance, the NLU engine provides the intention label and slot-value pairs.

Politeness Detection Module

Politeness detection is one of the crucial processes in the proposed DS architecture as it drives the conversational flow and robot behaviour. This module takes the input utterance and detects politeness using linguistic features ranging from 1 to -1 (very polite to very impolite). The RNN model with *sigmoid* function is used to learn the politeness from Stanford Politeness Corpus[2] (Danescu-Niculescu-Mizil et al., 2013). We fine-tune the trained model for the experimental dataset (mentioned in the previous section) that is created for the particular scenario to minimize uncertainty in prediction. For the sake of conceptual and computational simplicity, we discretized the politeness values into three categories: polite (1), neutral (0) and impolite (-1); see the examples in Table 7.2.

We use same RNN model as given in Figure 4.2(a) discussed in Chapter 4, except we use the *sigmoid* function instead of the *softmax* function and the word embeddings are learned during the training process. The model predicts politeness values in the range of 0 to 1; hence, the *sigmoid* function, then these values converted into the politeness classes such as:

DoP Range	0.0 - 0.4	0.4 - 0.6	0.6 - 1.0
Class	impolite (-1)	neutral (0)	polite (1)

Moreover, these discrete values are summed cumulatively over the conversation to calculate the degree of politeness and further used to modulate the dialogue

[2] https://www.cs.cornell.edu/~cristian/Politeness.html

flow of the proposed DS.

Additional Module

This module is open to adding additional socio-linguistic features such as sentiment or emotion. Adding more features can increase the complexity of the dialogue system, especially the dialogue flow module. However, it could be useful in some cases to incorporate multiple features and modalities to produce the required behaviour.

7.3.2 Dialogue Flow Module

The dialogue flow (DF) is a central engine of the system which communicates with most of the modules. It is implemented as a primary function to drive the DS, mainly connecting the user dialogue acts to the system dialogue acts. A rule-based and probabilistic belief tracking or dialogue state tracking model could be used to maintain the dialogue flow (Ultes et al., 2017). We used a rule-based model where the dialogue flow module keeps track of the user dialogue acts and DoP, extract a system dialogue act and send them to the response manager to fetch the appropriate responses.

The DF has a queue to store the context information of the preceding utterances to complete the state loop. It is useful to trigger the system dialogue acts from the response manager based on the context information and the current user dialogue act. For example, suppose the last user dialogue act is *TakeToPlace*. In that case, it triggers the *FinishedOne* system dialogue act to inform the user that the last action is finished providing a conditional flag. The system with the *FinishedOne* dialogue act asks if the user wishes to visit the next place. The contextual information in the loop keeps track whether the user accepts or rejects the proposal using the *Accept* and *Reject* user dialogue acts. If the *Accept* dialogue act appears, the robot takes the user to the next location until the list of locations is finished. If the user rejects with the *Reject* dialogue act, then the dialogue is ended with the conventional closing (*ConvClosing*) system dialogue act.

7.3.3 Response Management Module

The response manager is responsible for mapping the right response for the given intention and degree of politeness. In principle, the response manager maps the system dialogue acts and degree of politeness with the responses. Pre-defined response templates are stored in a data file that is accessed continuously during the interaction. Here are some examples of the stored responses:

```
"TakeToPlace": {
    "polite"  : {"op_utt": ["Please follow me, I can show you
                            the [slot_value]"]},
    "neutral" : {"op_utt": ["Please follow me, I can take you
                            to the [slot_value]"]},
    "impolite": {"op_utt": ["Please follow me.",
                            "Sure, follow me.",
                            "Sure."]}
},
"Thanking": {
    "polite"  : {"op_utt": ["It was my pleasure, you are welcome,
                            hope to see you again."]},
    "neutral" : {"op_utt": ["You are welcome, thanks for bearing
                            with me."]},
    "impolite": {"op_utt": ["You are welcome."]}
}
```

As we can see from the examples given above, when the DF module, consider, gets the utterance with the dialogue act *TakeToPlace* and *polite* the respective response (output utterance "op_utt") has to be fetched from the given templates. One can define several forms of the same utterance, as we can see for the *impolite* utterance of the *TakeToPlace* dialogue act, one of them is randomly or empirically picked to reduce the monotonousness in the responses. The slot-value pair names can be used in the output utterance by using *[slot_value]* field. This field gets filled with the respective value of the slot when they occur in a given conversational situation. Also notice the variations adapted for different politeness classes with the shortness of the response utterances, for example, in the *Thanking* dialogue act. Please see Appendix B.2 for more examples of the response templates used in this experiment.

7.4 Robot Navigation and Behavioural Control

7.4.1 Humanoid Robot Platform: Pepper

Pepper is a 1.2 meter tall omnidirectional wheeled humanoid robot platform. It is capable of exhibiting body language, perceiving and interacting with its surroundings, and move autonomously. Due to its 17 joints and 20 degrees of freedom kinematic configuration and edgeless design, the system is suitable for safe HRI (Pandey and Gelin, 2018). The platform is equipped with various sensors and actuators that ensure safe navigation and a high degree of expressiveness. LED's are distributed across the head (eyes and ears) and torso (shoulders) to support non-verbal communication by modifying colour and intensity. The microphones and speakers allow verbal interaction as well as environmental awareness. Sensing components include three laser sensors, two sonars and two infrared sensors located in the robots' base, two cameras, and a three-dimensional camera located in the head. In addition to them, two tactile sensors on the back of both hands allow human-robot physical awareness. Finally, the platform is powered by an Atom processor with a 1.91 GHz quad-core unit that allows the NAOqi SDK to orchestrate the different hardware elements as well as their access from other APIs, such as embedded speech recognition. Pepper is one of the widely used humanoid robots in HRI research experiments (Perera et al., 2017; Suddrey et al., 2018).

7.4.2 State Manager Module

In order to produce the physical and verbal responses in accordance with the degree of politeness exhibited during the interaction, a behavioural model has been designed in the State Manager, inspired by the valence and arousal model (Beck et al., 2010). The model is given with the discrete politeness values (1, 0, -1), computed from the last utterance, and degree of politeness (DoP) is stored as a cumulative sum in a sequential manner for every utterance in the given conversational interaction. The state manager maps the DoP as the cumulative sum of the previous and current utterances to different actuators. The actuators used to characterize the robots' change of behavioural state are the LED's colour (Nijdam, 2009), head pitch orientation (Lemaignan et al., 2016), voice pitch (Hubbard et al., 2017) and navigation speed, and they are mapped following the intuition

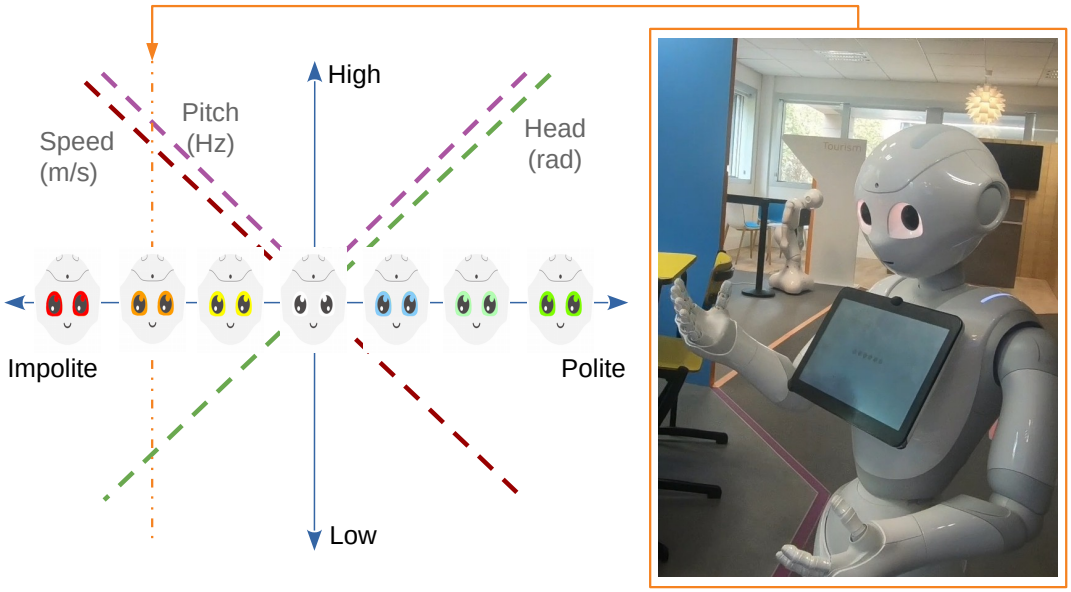


Figure 7.3: The behavioural model used to create the verbal and non-verbal responses based on the cumulative sum of the DoP. The Pepper robot shown in the right is in the position of the vertical orange line in the plot during the interaction.

as given in Figure 7.3. For example, the head pitch position raises as DoP increases, whereas the robot's navigation speed does the opposite. The voice pitch increases equally for both extreme politeness values and eyes colour code matches the robot's internal emotional state.

In this way, a multi-variability is provided to every single social cue that can vary in order to fit the interactive behavioural model. The user being repetitively polite during the whole interaction will experience a decrement in the robot's navigation speed. The head position orients towards the user, LEDs of eyes turn green with a slightly higher voice pitch (a similar state is depicted in Chapter 2 Figure 2.6). It is due to the fact that the degree of politeness has cumulatively incremented to a certain high positive number. However, it can not grow infinite, and hence we limit it to reach in the range of +3 to -3. That means we have seven individual states for each behavioural variable, as shown in Figure 7.3 and aside is a picture of the Pepper robot in one of the behavioural model states.

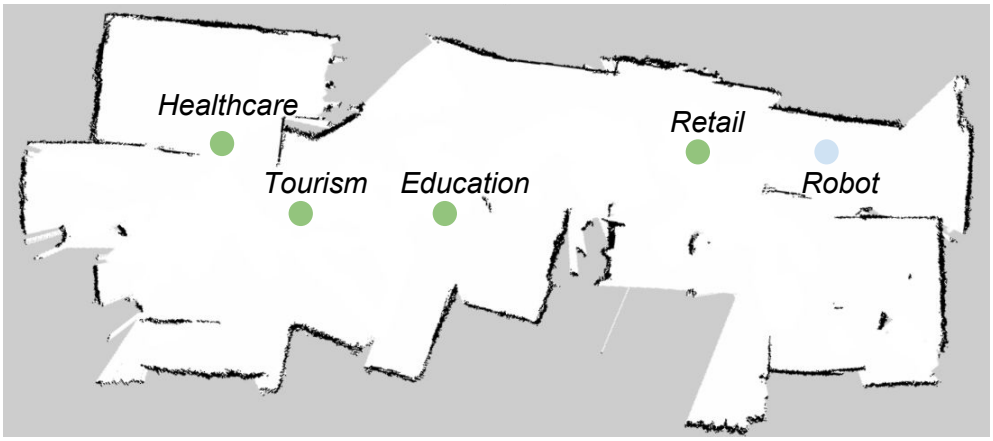


Figure 7.4: The environment map created with the Pepper robot and gmapping from ROS.

7.4.3 Motion Manager Module

The motion manager is responsible for navigation and planning, and adjusting speed according to the behavioural model explained in the last section. The navigation environment consists of the four locations to visit with the robot: Retail, Education, Tourism, and Healthcare, as shown on the map in Figure 7.4, The navigation can be operated in the following three modes: Tele-operation, Scripted Navigation and Navigation with Mapping.

Tele-operation

In this mode, the Pepper robot can be teleoperated by an operator sitting behind the computer. The NAOqi framework allows such a mechanism using the *moveToward* function from the ALMotion service, and the keys on the keyboard can be used for moving or stopping the robot. This method is not efficient; however, with this method, we could learn the location point distances from each other with reference to the initial position of the robot. These measurements are then used to calibrate the distances in the scripted navigation.

Scripted Navigation

The scripted navigation is achieved by commanding the robot to move to specific locations with the known distances in the environment. The distances can be measured with the help of teleoperation and some with manual measurements.

The *moveTo* command from the ALMotion service also allows adding a specific distance for the robot to be reached. We specify how far the robot has to move (in meters) and the orientations (in radians) it has to take during motion from the robot's initial position to each location in the map. The robot could be asked to reach any location, and it has to remember its current location to be able to know the distances from the other locations. We found this method very useful for the given simple navigation environment. It is easy to install for the known environment but not easily scalable to the unknown locations, hence the next method was introduced to map and plan the navigation autonomously.

Navigation Mapping and Planning

This method requires the use of the Robot Operating System (ROS), an open-source middle-ware framework. To fit our navigation need, we have adopted the following approach for generating and post-processing the map. The current readings of the Pepper's depth image sensor are converted into virtual laser data, using the package *depthimage_to_laserscan* (Perera et al., 2017; Suddrey et al., 2018). Hence, the map is generated using only the virtual laser data. An offline map (shown in Figure 7.4 is post-processed for testing purposes) can be acquired using *gmapping* (laser-based SLAM algorithm) (Grisettiyz et al., 2005). Then, the localization is performed using Adaptive Monte Carlo Localization (*acml*) (Fox, 2003). Finally, the navigation system uses a global planner with a map of inflated obstacles (costmap) and a local costmap with observations from the virtual laser data. The Dialog Flow requests a location from the API server (on the robot hosted with a virtual machine) using an ID (location name), and this one sends the coordinates to the ROS navigation stack to execute the path.

7.5 Results and Discussion

This experiment demonstrates the conversational dialogue system for social robots mainly driven by socio-linguistic cue politeness. The tour scenario consists of four departments in the lab: Retail, Education, Tourism and Healthcare, as shown on the map in Figure 7.4. The robot acts as a guide which takes the user to the particular requested department location using verbal interaction, as mentioned in Section 7.3.2. When a user requests the robot with the input utterance gets processed by the DA recognition module, which produces the result, as

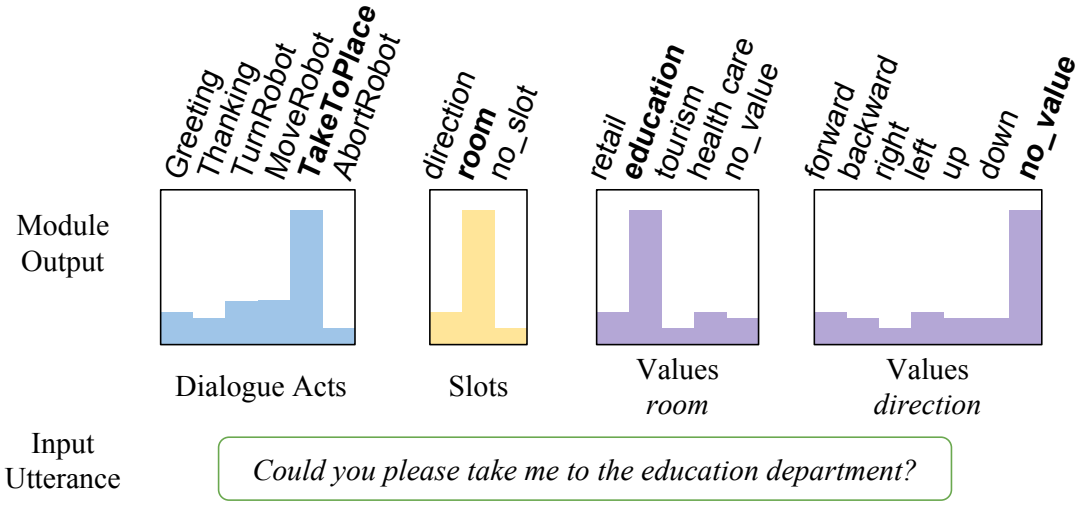


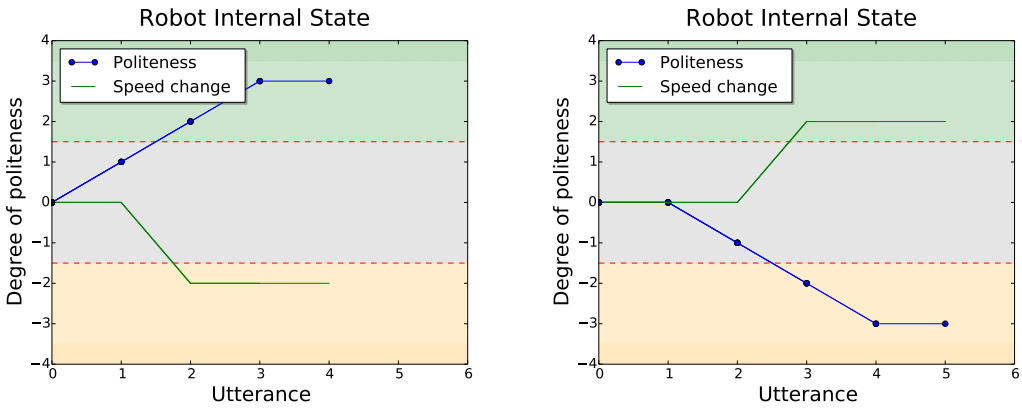
Figure 7.5: Output of the DA recognition module.

shown in Figure 7.5. The politeness detection module provides the politeness of that utterance. The dialogue flow sums the politeness values cumulatively over the conversation and communicates this information with all the following managers. Based on the degree of politeness, calculated as a cumulative sum of the politeness values, the robot adapts behavioural changes such as speeding up or down while navigating to the locations and changing the pitch of speech and the pitch angle of the head.

We tested this system on the Pepper robot with different users, expressing different levels of politeness. The behavioural variation and adaptation to speed change are based on the change in DoP. As can be seen, a few resulted in conversation plots given in Figure 7.6, for polite and impolite interaction scenarios. As we can see in part (a) of the figure, the DoP increases with a maximum of +3 (shown with blue line), and speed decreases (as shown with a green line), the corresponding log of dialogue is given in Table 7.3. The robot slows down and spends more time with the user when the user uses polite utterances. On the other hand, when the user uses impolite utterances, the robot speeds up and executes motion faster, as shown in part (b) of the figure. The corresponding dialogue log is given in Appendix Table B.2. The DoP changes gradually as it is cumulatively added and subtracted over the utterances in the conversation. Hence, in this dialogue system, DoP will not directly reach neutral (zero) or negative position, unless we can detect the intensity of the politeness similar to emotion intensity detection.

Turns	Utterances	Dialogue Act	PV	DoP
User	Hello how are you?	'Greeting', 1.0, {}	1	1
Robot	Hello, I am fine, thanks, how can I help you?			
User	Could you please show me the education department?	'TakeToPlace', 1.0, {['department', 'education']}	1	2
Robot	Please follow me, I can show you the education department.			
Robot	We arrived at the education department, here you can find the education related stuff. Would you like to visit next department?	'FinishedOne', {['department', 'education']}		
User	Yes I would like to visit	'Accept', 0.56, {}	1	3
Robot	Okay lets go to the next department.			
Robot	Please follow me, I can take you to the tourism department.	'TakeToPlace', 1.0, {['department', 'tourism']}		
Robot	We arrived at the tourism department, here you can find the tourism related stuff. Would you like to visit next department?	'FinishedOne', 1.0, {['department', 'tourism']}		
User	No thank you.	'Thanking', 1.0, {}	0	3
Robot	You are welcome, thanks for bearing with me.			

Table 7.3: Output example of the polite interaction.



(a) (b)

Figure 7.6: Robot internal state for (a) polite and (b) impolite interactions.

That would be an idea for the future work where we can infer the linguistic impatient from the user input utterance by using repetitive phrases to intensify the politeness. It will be necessary when we want the robot to switch from the slower to faster modes quickly.

The proposed behaviour of the robot for different situations, shown in the figure, is mainly to demonstrate the developed system and the efficacy of the proposed framework. The results indicate that the system is able to consider the linguistic features to modulate the navigation and behaviour of the robot in a coherent theoretical and functional framework. As aforementioned, to the best of our knowledge, such implementation of the framework is one of the first attempts of its kind. However, it is important to mention that the validation of the hypotheses about the most appropriate robot behaviours is not within the scope of this thesis, and it might require further investigation and user studies. Such studies are one of the next steps to utilize the framework for different scenarios, for example, inclusion of other socio-linguistic features into the natural language understanding module. The demonstration video and dialogue logs of the generated graphs in Figure 7.6 are available at the SECURE EU Project website: `https://secure-robots.eu/fellows/bothe/secondment-project/`.

7.6 Summary

We developed a dialogue-based navigation system for integrating intention and politeness features to the multivariate adaptation of the robot. We successfully deployed and tested our system on the Pepper humanoid robot with different levels of politeness. Currently, this experiment does not elicit the causal explanation for the behaviour and the multivariate adaptation of the robot. However, our experimental framework opens up a new challenge for studying the effect of politeness in human-robot verbal interaction scenarios. We firmly find this experiment useful in bridging the gap between socio-linguistic research and the HRI community. This research shall also help target the deployment of social-service robots with adaptation to socio-linguistic features such as politeness. In this work, the robotic behaviours are based on the previous research; however, they can be modified as per scenarios and requirements. This experiment uses politeness as a socio-linguistic cue such that being impolite tends to use short and direct commands as against to the longer and requests in polite interaction. That implies polite representing normal situation whereas impolite representing emergency. We successfully demonstrate its use in HRI experiment where the robot adapts this socio-linguistic changes in humans. As a result, we show how the robot can react in safety or emergencies using politeness socio-linguistic cues.

Chapter 8

Discussion and Conclusion

One of the identities of social robots or human-like agents is expected to exhibit natural language communication while maintaining safer interaction with humans. Natural language communication is one of the fascinating capabilities of humans that is developed with high complexity. While it is reasonable to realize this existing human ability to interact with the robots, learning from the humans' conversational behaviours becomes crucial. The focus of this thesis is to join the effort from the natural language processing research towards safe human-robot interaction using conversational analysis and computing techniques. The main goal is to contribute to the knowledge of understanding the socio-linguistic features such as emotion, politeness and dialogue act, towards building a natural language understanding for safe human-robot interaction.

8.1 Thesis Summary

The thesis discovers essential facts in natural language understanding and conversational analysis to address the goal mentioned above, formulating narrow research questions on conversational language learning and safe human-robot interaction. In brief, understanding the humans' behavioural changes in verbal communication can enable us to transfer that knowledge into social robots. We present the developed methods and models that allow performing language learning for conversational analysis. We explore the recurrent neural network approaches used in the learning process of dialogue acts, emotions and politeness. These methods later provide the ability to be used in the dialogue system, developed for safe human-robot interaction scenarios.

Dialogue acts aids conversation system to produce a natural dialogue flow; however, it is unclear how other socio-linguistic features such as emotion and politeness affect the conversational flow in HRI. Context-based learning improves the performance of the dialogue act or emotion recognition, but we answer how many utterances in the context are useful; how do they contribute to the final result. The first model, a context-based model using recurrent neural networks (RNN), helps to discover that only a few preceding utterances are sufficient to recognize the dialogue act of the current utterance. The second model, a context-based utterance-level attention mechanism on top of the bidirectional RNN model, extends the contextual learning of the preceding utterances by providing an ability to compute each utterance's contribution in the context.

Furthermore, we apply similar techniques to contextual emotion detection and found that they exhibit the same properties to the dialogue acts in the conversational analysis. We deploy neural ensemble modelling for emotion recognition in the conversation and found that end-to-end models perform better than domain-specific lexicon-based models. We also found that how dialogue-based learning to estimate the next utterance's sentiment helps discover undesirable or unsafe situations during the conversational interaction. We propose a new paradigm, the emotional dialogue acts, annotating multi-modal conversational emotion datasets with the dialogue acts. We created an ensemble of neural annotators trained on the dialogue act corpus and used it to annotate the emotion corpora. We have shown how some dialogue acts and emotions commonly occur together in many instances, such as apology with sadness, thanking with happiness, and rejection with frustration.

Finally, we demonstrate the use of socio-linguistic feature, politeness, in an HRI scenario, by considering the user's safety and desirability. For example, if the user utters short sentences or finds user utterances linguistically impolite, the robot acts quicker and faster. In contrast, the same phenomenon could occur if the user uses polite language and indicates that the user wants to spend more time with the robot. The experiments and models presented in the thesis allow transferring the knowledge learned through them, confirming the hypothesis made on the language learning during research question formulations, and supporting the pose of novel suggestions to the research community towards safe human-robot interaction.

8.2 Discussion

We performed a series of experiments to prove the hypothesis and research questions posed during the exploration of this thesis work. As mentioned already, this thesis mainly focuses on the natural language understanding and conversational analysis that helps to develop a dialogue system for HRI to adapt different robotic actions based on the behavioural changes found in the human communication. We will discuss the thesis by answering the research questions posed in the introduction chapter:

Question 1: How can we find the number of preceding utterances in the context that are required towards recognizing the dialogue act of the given current utterance?

During the development of the concept of a dialogue system for HRI, we primarily looked into the dialogue acts (DA) of the input utterances. We performed a conversational analysis and found that context-based learning has to be applied in the dialogue act recognition task. However, only past utterances can be used because technically any dialogue system can access only the past utterances, that applies to the HRI scenarios. In our experiment, presented in Section 4.4, we found that only three utterances in the context are sufficient to produce better accuracy over all the test set of the SwDA corpus. This experiment achieved state-of-the-art accuracy on the given test set; however, this experiment could not help to find out how to investigate the particular set of utterances to answer the mentioned question. Hence, we conducted another experiment, presented in Section 4.5, where an utterance-level attention-based bidirectional RNN model is used to compute the contributions of each utterance in the context. This way, we discover that two preceding utterances in the context contribute in higher amount during the dialogue act recognition of the current utterance (which is presented graphically in Figure 4.8).

During this study, we found that different dialogue acts behave differently due to their contextual differences. For example, if there is an *answer* DA utterance, the previous sentences might contain a *question* DA utterance which will substantially contribute towards recognition. However, if there is a reverse case, then the contribution of the past utterances (*answer*) can be negligible when predicting the current utterance (*question*). That led to asking the next question, which was also posed in the research community but was not answered experimentally:

Question 2: How much does each utterance in the context contribute towards recognizing the dialogue act of the given utterance?

As mentioned already while answering the previous question, the experiment presented in Section 4.5, the attention mechanism is used to compute the weights over a set of the input utterances, as a result eliciting the amount of contribution of each utterance. In this experiment, we took the sets of five utterances (batches of size five), and process them with BiRNNs along with the attention layer which computes the weights for every utterance in the batch. We discover that the preceding two (at the most three) contributes substantially towards the current utterance. It is due to the fact that most recent utterances contain precious context as against to the very past utterances for the dialogue act recognition. This way, we successfully show the methods to discover the required number of utterances in the context and to compute their contribution towards the dialogue act recognition.

We learn to know others feelings via emotion expressions, and we found that the identifying emotion in the conversation is a challenging task, especially, when other modalities are absent such as facial expressions or sound variations. We experiment with contextual emotion recognition in conversation and learn about their importance in the decision-making process. For example, the extreme sentiment polarities in the utterances of conversation may convey negativeness or positiveness in the context. Hence, the next research question posed was:

Question 3: How can dialogue-based neural learning estimate the sentiment of the next utterance help us find undesirable events or safety-critical cues for safe human-robot interaction?

To answer this question, we proposed to use a dialogue-based context learning model to estimate the sentiment of next utterance. This model uses a special recurrent neural network, long short-term memory network, that takes preceding utterances as a context to predict the sentiment of the next utterance. The concept is derived from the feedback learning phenomenon; for example, appreciation or desirable moments are usually expressed with positive sentiment, whereas negative sentiment is expressed on undesirable or unsafe or unhappy moments. These extreme sentiment utterances are used as feedback cues, while their preceding utterances are providing the context. We successfully show that the model learns to estimate the next utterance sentiment; the experiment is presented in Section 5.4. We deeply analyze the predicted sentiment values in the test set and dis-

cover that the results show an interesting phenomenon of predicting undesirable or safety-critical situations. For example, in a particular scenario, on informing through the utterances that the "chair is broken", the model raises a negative sentiment. This experiment supports proving the hypothesis and answering the research question; however, it requires building a proper foundation of sentiment analysis in conversations. On the other hand, it is also an indirect elicitation based only on the sentiment polarities, though we use fine-grain numerical values. Hence, adequate improvement is required before its integration in the HRI scenarios. However, this experiment is a step towards language learning for safe human-robot interaction.

Emotional dialogue acts, as presented in Section 6.3, the co-occurrence relations between emotion expressions and dialogue acts, provide interesting discovery; as a result, to answer the following question:

Question 4: How can we reliably use the neural ensemble method to enrich existing emotion data with dialogue act labels? Do the emotions and dialogue acts provide any relations among themselves that would be useful to consider for the conversational analysis?

As presented in Chapter 6, we show how effectively the ensemble model of the neural annotators annotates the conversational emotion data for the dialogue act labels. As emotion and dialogue acts are considerably different aspects of the language, we found that their inter-relationship brings another dimension to the natural language understanding. This analysis of the relationships between them leads to exploring its incorporation into the dialogue systems. It is one of the primary goals, as we stated that different socio-linguistic features for the in-depth conversational analysis could lead us to a better understanding of the human-human interaction. We designed a dialogue system that incorporates politeness as a socio-linguistic feature into the natural language understanding module that drives the conversational flow and helps to answer the next research question:

Question 5: How to combine the socio-linguistic features such as emotion or politeness with the dialogue acts in the dialogue system for HRI? How does that help to influence the output behaviour of the robots?

In Chapter 7, we demonstrated the dialogue-based navigation system that combines the politeness feature with the dialogue acts to produce different behaviours and actions from the robot. We find that using such features to modulate

robotic actions and behaviours make robots more autonomous and reliable. This additional dimension to the natural language understanding module makes it possible to reach a higher level of autonomy, as discussed in Section 2.5. Politeness comprehension makes the robot distinguish between different interactive behaviours such as order versus request instruction, urgent versus normal event, and patient versus impatient user. It provides a basis to our ultimate goal to make use of the knowledge gained with these experiments and analysis for safe human-robot interaction.

With the proposed methods and approaches, we achieve fundamental knowledge to make it possible to utilize different socio-linguistic cues for safe human-robot interaction. The experiment of learning desirable or safety-critical situations from the conversational language brings a higher dimension to the language understanding for HRI. As the robot would be able to anticipate dangerous situations and react accordingly to avoid a possible hazard, it can improve the trust of the robot in society. The robot can be aware of possible changes in the environment and the safety-critical situations; it can inform the humans around verbally. We are aware that detecting the safety-related cues as early as possible might produce false-positives; however, they can be accepted (or quickly resolved through a query within the dialogue) if possible dangers can be avoided when they occur in the given scenario. In another experiment, we propose to use politeness as a social cue to understand users patient or impatient behaviour. As we know, when a user is impatient or in an urgency situation, we use short sentences and orders instead of requests. The robots being aware of such politeness comprehension, make them more acceptable in society. On the other hand, in a possibly unsafe situation, we expect the robots to react quicker, which is possible if the robot is aware of the safety-critical situations and understands users' urgency through the language learning.

8.3 Limitations and Future Work

To comprehensively identify the most suitable natural language understanding for human-robot interaction, the methods and approaches explored in this thesis can be modified in several ways. All the approaches and models proposed in this thesis are based on the textual language processing alone; however, this can be addressed in the future work by adding multiple modalities such as speech

or vision-based socio-linguistic feature detection. It is needless to mention that emotion or politeness is perceived better with vision and speech in addition to textual processing in conversation. On the other hand, the politeness comprehension model was designed intuitively by studying the literature for the given human-robot interaction scenario, which could be extended as a neural model to learn from conversational interactions. These effect of the socio-linguistic features in human-robot interaction can be considered for performing the user studies for better understanding of such linguistic features.

Conversational interaction occurs in an incremental fashion, and the human input utterances can always be similar but with variance due to the use of natural language. The proposed neural models cannot be directly applied for online or incremental learning, as they are trained with gradient descent mechanism for finding the best weight setting on the given datasets. Hence, future research must address the possibility to continue the training of the model with minimal changes in the parameters.

8.4 Conclusion

This thesis contributes to understanding socio-linguistic features such as emotion, politeness and dialogue act, towards building a natural language understanding for safe human-robot interaction. Combined comprehension of social cues and linguistic features can integrate novel experience and safety during human-robot verbal interaction. With the knowledge about conversational analysis and neural modelling to perform in-depth experimentation for language learning, we can design human-like artificial agents, that are aware of the environment and interact with safety concerns with humans. That will help to realize safer social robots with verbal communication capability.

Appendix A

Publications and Associated Activities

A.1 List of Publications Associated this Thesis

Following publications are related to the research contribution of this thesis, published as a part of **SECURE** (Safety Enables Cooperation in Uncertain Robotic Environments) EU Project, hosted by the **University of Hamburg**, where this thesis primarily contributes. This project is funded by the EU Horizon 2020 research and innovation programme under grant agreement No 642667 (`http://secure-robots.eu`). As per EU Project guidelines, all the publications are available and accessible at: `https://secure-robots.eu/fellows/bothe/` and `https://www.inf.uni-hamburg.de/en/inst/ab/wtm/people/bothe`.

P2020 : **Bothe, C.**, Weber, C., Magg, S., and Wermter, S. (2020). EDA: Enriching Emotional Dialogue Acts using an Ensemble of Neural Annotators. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2020*, pages 620–627. European Language Resources Association (ERLA).

P2019 : **Bothe, C.** and Wermter, S. (2019). SemEval-2019 Task 3: Ensemble BiRNNs for Contextual Emotion Detection in Dialogues. In *Proceedings of the International Workshop on Semantic Evaluation (SemEval-2019) at the Conference NAACL-HLT 2019*, pages 261–265. Association for Computational Linguistics (ACL).

P2018a : **Bothe, C.**, Garcia, F., Cruz-Maya, A., Pandey, A. K., and Wermter, S. (2018). Towards Dialogue-Based Navigation with Multivariate Adaptation Driven by Intention and Politeness for Social Robots. In *Proceedings of the International Conference on Social Robotics, ICSR 2018*, pages 230–240. Springer International Publishing.

P2018b : **Bothe, C.**, Magg, S., Weber, C., and Wermter, S. (2018). Conversational Analysis using Utterance-level Attention-based Bidirectional Recurrent Neural Networks. In *Proceedings of the International Conference INTERSPEECH 2018*, pages 996–1000. International Speech Communication Association (ISCA).

P2018c : **Bothe, C.**, Magg, S., Weber, C., and Wermter, S. (2018). Discourse-Wizard: Discovering Deep Discourse Structure in your Conversation with RNNs. *Computation and Language*.

P2018d : **Bothe, C.**, Weber, C., Magg, S., and Wermter, S. (2018). A Context-based Approach for Dialogue Act Recognition using Simple Recurrent Neural Networks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018*, pages 1952–1957. European Language Resources Association (ERLA).

P2018e : Zhou X., Weber C., **Bothe C.**, and Wermter S. (2018). Hybrid Planning Strategy through Learning from Vision for Target-directed Navigation. In *Proceedings of the International Conference on Artificial Neural Networks, ICANN 2018*, pages 304–311. Springer International Publishing.

P2017a : **Bothe, C.**, Magg, S., Weber, C., and Wermter, S. (2017). Dialogue-based neural learning to estimate the sentiment of a next upcoming utterance. In *Proceedings of the 26th International Conference on Artificial Neural Networks, ICANN 2017*, pages 477–485. Springer International Publishing.

P2017b : Lakomkin, E.*[1], **Bothe, C.***[1], and Wermter, S. (2017). GradAscent at EmoInt-2017: Character and Word Level Recurrent Neural Network Models

[1]equal contribution

for Tweet Emotion Intensity Detection. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis at the Conference EMNLP 2017*, pages 169–174. Association for Computational Linguistics (ACL).

A.2 Secondment Project

One month secondment was conducted from 25th July 2018 to 24th August 2018, as a part of the SECURE project at the Innovation Department of SoftBank Robotics Europe, Paris, under the supervision of Dr Amit Kumar Pandey, Head Principal Scientist (now President and Chief Technology Officier at Hanson Robotics) with Fernando Garcia and Dr. Arturo Cruz-Maya. **SoftBank Robotics Europe** is an associated industrial partner of the **SECURE** EU Project, along with the involvement of **CROWDBOT** and **MUMMAR** EU Projects.
The outcomes, produced almost in the same month, are a dialogue-based navigation system driven by politeness for a social robot tested on the Pepper humanoid robot, a video demo (link provided in the following section) and an article (P2018a) titled "Towards Dialogue-Based Navigation with Multivariate Adaptation Driven by Intention and Politeness for Social Robots" published in the proceedings of and presented at the *International Conference on Social Robotics, ICSR 2018* in Quingao, China with the video demo available at:
`http://secure-robots.eu/fellows/bothe/secondment-project/`

A.3 Conference and Workshop Organizations

ICDL-EpiRob 2019 Workshop on Personal Robotics and Secure Human-Robot Collaboration (Joint APRIL and SECURE ITN Symposium) This jointly organized workshop by the **SECURE** and **APRIL** EU Project fellows focusing on learning and interaction-based approaches to safe human-robot collaboration and their application to personal robotics. The workshop was co-located with *ICDL-EpiRob 2019*. The workshop included research topics like: developmental approaches to robot learning, safe interaction in uncertain environments, affect and emotion modelling for safe human-robot interaction, language and non-verbal communication etc.

https://secure-robots.eu/icdl2019-workshop/programme/
https://icdl-epirob2019.org/workshops/

The *International PhD Conference on Safe and Social Robotics (SSR-2018)*: Two EU Horizon2020 Projects **SOCRATES** and **SECURE** with the main focus on Robotics are jointly organized this conference. The conference was co-located with *IROS 2018* in Madrid, Spain. It was an opportunity for young researchers and PhD students in Human Robot Interaction (HRI) to showcase their research and connect with other researchers, fellows and acclaimed senior researchers in Robotics, Social Sciences, Machine Learning and Assisted Living to name a few. http://www.socrates-project.eu/sesoro-2018/

Peer Network Workshop "Project Review" (2017): Workshop organization by the fellow at the University of Hamburg during the Mid Term Review meeting. Program was chaired by Chandrakant Bothe, Mahammad Ali Zamani, Egor Lakomkin. The workshop was organized to train the fellows on projects and reviews with several practical sessions such as Collaborative Project Search, Review Collaborative Projects, Discussion and Ranking of Collaborative Projects, and Fellow Progress Presentations and Discussion. This workshop was then followed by the Mid Term Review Meeting of SECURE EU Project in May, 2017. Lecture highlights: Prof. Stefan Wermter delivered an "Interactive Lecture: Projects and Reviews" and Dr. Sven Magg delivered an "Interactive Lecture on Progress, Review Process and Outcomes".

A.4 Corpora and Demonstration Links

Emotional Dialogue Acts (EDA) Corpora: Enriching Existing Conversational Emotion Datasets with Dialogue Acts using Neural Annotators (P2020). Dialogue Act Recognition Demonstration server API which has 3 with context models and 2 without context models. The API information is available at:
https://github.com/bothe/EDAs

Discourse-Wizard Web Demo: Discovering Deep Discourse Structure in your Conversation with RNNs (P2018b), also refer to (P2018c; P2018d).

Dialogue Act Recognition Demonstration with and without context model, shows the importance of context in a conversation. The live web-demo is available at :
`https://secure-robots.eu/fellows/bothe/discourse-wizard-demo/`
`https://bothe.github.io/discourse-wizard/`

Secondment Project Video Demo: Dialogue-based Navigation with Multivariate Adaptation driven by Intention and Politeness for Social Robots (P2018a). Video demo of the secondment work accomplished in collaboration with SoftBank Robotics Europe in Paris, France during July-August 2018.
`https://secure-robots.eu/fellows/bothe/secondment-project/`

Appendix B

Additional Notes

B.1 Switchboard Dialogue Act Corpus Statistics and Tag Set

Table B.1: Statistics of SwDA corpus with the Dialogue Act tags, and their SWBD-DAMSL names with examples (Stolcke et al., 2000).

SWBD-DAMSL	SWBD	Example	Count	%
Statement-non-opinion	sd	Me, I'm in the legal department.	72,824	36%
Acknowledge (Backchannel)	b	Uh-huh.	37,096	19%
Statement-opinion	sv	I think it's great	25,197	13%
Agree/Accept	aa	That's exactly it.	10,820	5%
Abandoned or Turn-Exit	% -	So, -	10,569	5%
Appreciation	ba	I can imagine.	4,633	2%
Yes-No-Question	qy	Do you have to have any special training?	4,624	2%
Non-verbal	x	[Laughter], [Throat_clearing]	3,548	2%
Yes answers	ny	Yes.	2,934	1%

Conventional-closing	fc	Well, it's been nice talking to you.	2,486	1%
Uninterpretable	%	But, uh, yeah	2,158	1%
Wh-Question	qw	Well, how old are you?	1,911	1%
No answers	nn	No.	1,340	1%
Response Acknowledgement	bk	Oh, okay.	1,277	1%
Hedge	h	I don't know if I'm making any sense or not.	1,182	1%
Declarative Yes-No-Question	qy^d	So you can afford to get a house?	1,174	1%
Other	o,fo,bc, by,fw	Well give me a break, you know.	1,074	1%
Backchannel in question form	bh	Is that right?	1,019	1%
Quotation	^q	You can't be pregnant and have cats	934	.5%
Summarize/reformulate	bf	Oh, you mean you switched schools for the kids.	919	.5%
Affirmative non-yes answers	na,ny^e	It is.	836	.4%
Action-directive	ad	Why don't you go first	719	.4%
Collaborative Completion	^2	Who aren't contributing.	699	.4%
Repeat-phrase	b^m	Oh, fajitas	660	.3%
Open-Question	qo	How about you?	632	.3%
Rhetorical-Questions	qh	Who would steal a newspaper?	557	.2%

Hold before answer/agreement	^h	I'm drawing a blank.	540	.3%
Reject	ar	Well, no	338	.2%
Negative non-no answers	ng,nn^e	Uh, not a whole lot.	292	.1%
Signal-non-understanding	br	Excuse me?	288	.1%
Other answers	no	I don't know	279	.1%
Conventional-opening	fp	How are you?	220	.1%
Or-Clause	qrr	or is it more of a company?	207	.1%
Dispreferred answers	arp,nd	Well, not so much that.	205	.1%
3rd-party-talk	t3	My goodness, Diane, get down from there.	115	.1%
Offers, Options Commits	oo,cc,co	I'll have to check that out	109	.1%
Self-talk	t1	What's the word I'm looking for	102	.1%
Downplayer	bd	That's all right.	100	.1%
Maybe/Accept-part	aap/am	Something like that	98	<.1%
Tag-Question	^g	Right?	93	<.1%
Declarative Wh-Question	qw^d	You are what kind of buff?	80	<.1%
Apology	fa	I'm sorry.	76	<.1%
Thanking	ft	Hey thanks a lot	67	<.1%

B.2 Examples of Response Templates

The template is arranged in the JavaScript Object Notation (JSON) format:

```
"DialogueAct": {
    "politeness"  : {"op_utt": ["utterance_1", "utterance_2",...]}
    "politeness"  : {"op_utt": ["utterance_1", "utterance_2",...]}
    ...
    }
```

where "DialogueAct" is a customized dialogue acts for example "FinishedOne", "Accept" or "Reject". "politeness" can be of the three classes: polite, neutral and impolite. "op_utt" contains a list of utterances, with variants such as "utterance_1", "utterance_2",...

```
"FinishedOne": {
    "polite"  : {"op_utt": [
                "We arrived at [slot_value] department, here you
                can find the [slot_value] related stuff.
                Let me know if you wish to visit next department?",
                "Here is the [slot_value] department, here
                you can find the stuff related to [slot_value].
                Do you wish to visit next department?" ]},
    "neutral" : {"op_utt": [
                 "We arrived at [slot_value] department,
                 here you can find the [slot_value] related
                 stuff. Would you like to visit next department?",
                 "Here is the [slot_value] department,
                 here you can find the stuff related to [slot_value].
                 Do you wish to visit next department?" ]},
    "impolite": {"op_utt": [
                 "We arrived at [slot_value] department.",
                 "Here is the [slot_value] department." ]}
},
"Accept": {
        "polite"  : {"op_utt": [
                    "Okay, I will direct you to the next department."]},
```

```
        "neutral" : {"op_utt": [
                    "Okay lets go to the next department."]},
        "impolite": {"op_utt": [
                     "Okay lets go to the next department."]}
},
"Reject": {
        "polite"  : {"op_utt": [
                   "Okay, enjoy the time."]},
        "neutral" : {"op_utt": [
                    "okay, have a nice day."]},
        "impolite": {"op_utt": [
                     "Okay. " ]}
},
"AbortRobotAction": {
        "polite"  : {"op_utt": [
                        "Oh, Something must be went wrong,
                        let me finish reconfiguring and
                        you can come back in some time." ]},
        "neutral": {"op_utt": [
                        "Wait, something went wrong, let me
                        reconfigure the actions." ]},
        "impolite": {"op_utt": [
                        "Sorry, something went wrong." ]}
},
"ConvClosing": {
        "polite"  : {"op_utt": [
                        "It was nice meeting you, have a nice day." ]},
        "neutral" : {"op_utt": [
                        "Bye bye, have a nice day." ]},
        "impolite": {"op_utt": [
                        "Bye bye." ]}
}
```

B.3 Output Example of the Impolite Dialogue

Turns	Utterances	Dialogue Act	PV	DoP
User	Hello	'Greeting', 1.0, {}	1	1
Robot	Hello, how can I help you?			
User	Show me the education department!	'TakeToPlace', 1.0, {['department', 'education']}	-1	-1
Robot	Please follow me.			
Robot	Here is the education department. Do you wish to visit next department?	'FinishedOne', {['department', 'education']}		
User	Show me the tourism department!	'TakeToPlace', 1.0, {['department', 'tourism']}	-1	-2
Robot	Please follow me.			
Robot	We arrived at tourism department. Would you like to visit next department?	'FinishedOne', 1.0, {['department', 'tourism']}		
User	No	'Reject', 1.0, {}	-1	-3
Robot	Okay.			
User	Bye.	''ConvClosing', ', 1.0, {}	-1	-3
Robot	Bye bye, have a nice day.			

Table B.2: Output example of the impolite interaction.

Appendix C

Acknowledgements

First, I would like to express my gratitude to all who directly and indirectly contributed to my PhD studies and encouraged me to complete.

Foremost, I sincerely want to acknowledge and thank my supervisor Prof. Dr. Stefan Wermter, for his constant support and guidance, consistent motivation and insightful advice from time to time. While going through different ups and downs of doctoral work, his wise suggestions helped me understand several aspects of research and personal life.

I also like to thank Dr. Sven Magg and Dr. Cornelius Weber for their consistent help in improving the experiments, writing, and problem formulation; and their thought and ideas shared via discussions on various occasions. Additionally, I especially like to thank Prof. Dr. Chris Biemann and Prof. Dr. Wolfgang Menzel for being a crucial part of the examination commission. Their thorough evaluation of my thesis helped me improve the overall message delivered in the thesis. I am grateful to Prof. Dr. Angelo Congelosi, Dr. Amit Kumar Pandey, Prof. Dr. Chu Kiong Loo, Dr. Johannes Twiefel, Dr. Matthias Kerzel, Dr. Burhan Hafez, Dr. Pablo Barros, Dr. Francisco Cruz, Dr. Nicols Navarro-Guerrero, Dr. Sascha Griffiths, Dr. German Parisi, Dr. Stefan Heinrich, Dr. Di Fu, Xiaomao Zhou, Dr. Manfred Eppe, Dr. Doreen Jirak for occasional vivid discussions both professionally and personally. I want to extend my thanks to Erik Strahl, Katja Köster, and Annegret Immer for technical and administrative support in my work at various stages.

It is my honour to acknowledge the University of Hamburg and SECURE EU Project's support and truly experience proud to be part of them. My friends and colleagues from SECURE Project at the University of Hamburg, Egor Lakomkin

and Mohammad Ali Zamani, shared the office with me and helped me learn and grow together with them. I like to thank them for the patience they showed to advance together. As a part of SECURE Project, I spend one-month secondment (internship) at Softbank Robotics, Paris. I want to thank Dr. Amit Kumar Pandey, Fernando Garcia and Dr. Arturo Cruz Maya for all the guidance and help during the experiments conducted at their facility.

Finally, I heartily thank my father and mother; they made me a person I am today. My father led me to start but unfortunately could not see the end of this insightful journey of my life. I love to thank my young brother Shrikant who was always supportive during this journey.

I could not have accomplished this work without a special person who was always on my side, my wife Anu (Sunanda). Together, we encountered several swings during this journey, and I admire your love, sensibility, and consistent trust in me. Special thanks for getting me to earn a father's title with a lovely daughter Parinidhi, born during the same year of finishing this work.

Bibliography

Adam, C., Johal, W., Pellier, D., Fiorino, H., and Pesty, S. (2016). Social Human-Robot Interaction: A New Cognitive and Affective Interaction-Oriented Architecture. In *Proceedings of the International Conference on Social Robotics*, pages 253–263. Springer.

Allen, B. (1993). The Historical Discourse of Philosophy. *Canadian Journal of Philosophy*, 23(sup1):127–158.

Allen, J. F., Byron, D. K., Dzikovska, M., Ferguson, G., Galescu, L., and Stent, A. (2001). Towards Conversational Human-Computer Interaction. *AI Magazine*, 22(4):27.

Allen, J. F. and Core, M. (1997). Draft of DAMSL: Dialogue Act Markup in Several Layers. *Carnegie Mellon University*.

Asimov, I. (1963). *I, Robot*. New York: Doubleday.

Aubakirova, M. and Bansal, M. (2016). Interpreting Neural Networks to Improve Politeness Comprehension. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2035–2041. Association for Computational Linguistics.

Austin, J. L. (1962). *How to Do Things with Words*. Oxford University Press.

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010*, pages 2200–2204.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural Machine Translation by Jointly Learning to Align and Translate. In *Proceedings of the International Conference on Learning Representations*.

Bastianelli, E., Castellucci, G., Croce, D., Iocchi, L., Basili, R., and Nardi, D. (2014). HuRIC: a Human Robot Interaction Corpus. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2014*, pages 4519–4526.

Beck, A., Cañamero, L., and Bard, K. A. (2010). Towards an Affect Space for robots to display emotional body language . In *Proceedings of the 19th International Symposium in Robot and Human Interactive Communication*, pages 464–469.

Beer, J. M., Fisk, A. D., and Rogers, W. A. (2014). Toward a Framework for Levels of Robot Autonomy in Human-Robot Interaction. *Journal of Human-Robot Interaction*, 3(2):74–99.

Berg, M. M. (2015). NADIA: A Simplified Approach Towards the Development of Natural Dialogue Systems. In *International Conference on Applications of Natural Language to Information Systems*, pages 144–150. Springer.

Bergman, C. and Davidson, J. (2005). Unitary embedding for data hiding with the SVD. In *Security, Steganography, and Watermarking of Multimedia Contents VII*, volume 5681, pages 619–631. International Society for Optics and Photonics.

Biswas, S., Chadda, E., and Ahmad, F. (2015). Sentiment Analysis with Gated Recurrent Units. *Advances in Computer Science and Information Technology (ACSIT)*, 2(11):59–63.

Bod, R. (2013). *A New History of the Humanities: The Search for Principles and Patterns from Antiquity to the Present*. Oxford University Press.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bothe, C. (2015). Human-Humanoid Interaction by Verbal Dialogue. Master's thesis, Ecole Centrale de Nantes, France.

Bothe, C., Garcia, F., Cruz-Maya, A., Pandey, A. K., and Wermter, S. (2018a). Towards Dialogue-based Navigation with Multivariate Adaptation driven by Intention and Politeness for Social Robots. In *Proceedings of the International Conference on Social Robotics*, pages 230–240. Springer.

Bothe, C., Magg, S., Weber, C., and Wermter, S. (2017). Dialogue-based neural learning to estimate the sentiment of a next upcoming utterance. In *Proceedings of the 26th International Conference on Artificial Neural Networks*, pages 477–485. Springer.

Bothe, C., Magg, S., Weber, C., and Wermter, S. (2018b). Conversational Analysis using Utterance-level Attention-based Bidirectional Recurrent Neural Networks. In *Proceedings of the International Conference INTERSPEECH 2018*, pages 996–1000. International Speech Communication Association (ISCA).

Bothe, C., Magg, S., Weber, C., and Wermter, S. (2018c). Discourse-Wizard: Discovering Deep Discourse Structure in your Conversation with RNNs. *Computation and Language*.

Bothe, C., Weber, C., Magg, S., and Wermter, S. (2018d). A Context-based Approach for Dialogue Act Recognition using Simple Recurrent Neural Networks. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018*, pages 1952–1957. European Language Resources Association (ERLA).

Bothe, C., Weber, C., Magg, S., and Wermter, S. (2020). EDA: Enriching Emotional Dialogue Acts using an Ensemble of Neural Annotators. In *Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020*, pages 620–627. European Language Resources Association (ELRA).

Bothe, C. and Wermter, S. (2019). MoonGrad at SemEval-2019 Task 3: Ensemble BiRNNs for Contextual Emotion Detection in Dialogues. In *Proceedings of the International Workshop on Semantic Evaluation (SemEval-2019) at the Conference NAACL-HLT 2019*. Association for Computational Linguistics.

Breiman, L. (1996). Bagging Predictors. *Machine learning*, 24(2):123–140.

Brennan, S. E. (2000). Processes that Shape Conversation and their Implications for Computational Linguistics. In *Proceedings of the 38th Annual Meeting on*

Association for Computational Linguistics, pages 1–11. Association for Computational Linguistics.

Brown, P. and Levinson, S. C. (1987). *Politeness: Some Universals in Language Usage*, volume 4. Cambridge University Press.

Brown, P. F., Desouza, P. V., Mercer, R. L., Pietra, V. J. D., and Lai, J. C. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Busso, C., Bulut, M., Lee, C.-C., Kazemzadeh, A., Mower, E., Kim, S., Chang, J. N., Lee, S., and Narayanan, S. S. (2008). IEMOCAP: Interactive emotional dyadic motion capture database. *Language Resources and Evaluation*, 42(4):335–359.

Castro-González, Á., Castillo, J. C., Alonso-Martín, F., Olortegui-Ortega, O. V., González-Pacheco, V., Malfaz, M., and Salichs, M. A. (2016). The Effects of an Impolite vs. a Polite Robot Playing Rock-Paper-Scissors. In *Proceedings of the International Conference on Social Robotics*, pages 306–316. Springer.

Chatterjee, A., Narahari, K. N., Joshi, M., and Agrawal, P. (2019). SemEval-2019 Task 3: EmoContext: Contextual Emotion Detection in Text. In *Proceedings of The 13th International Workshop on Semantic Evaluation (SemEval-2019)*, pages 39–48.

Chen, S.-Y., Hsu, C.-C., Kuo, C.-C., Huang, T.-H., and Ku, L.-W. (2018a). EmotionLines: An Emotion Corpus of Multi-Party Conversations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018*, pages 1597–1601.

Chen, Z., Yang, R., Zhao, Z., Cai, D., and He, X. (2018b). Dialogue Act Recognition via CRF-Attentive Structured Network. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 225–234. ACM.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. In *Proceedings*

of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1724–1734. Association for Computational Linguistics.

Chorowski, J. K., Bahdanau, D., Serdyuk, D., Cho, K., and Bengio, Y. (2015). Attention-Based Models for Speech Recognition. In *Proceedings of the Conference on Advances in Neural Information Processing Systems*, pages 577–585.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. In *NIPS 2014 Workshop on Deep Learning*.

Clark, E. V. (1978). Awareness of Language: Some Evidence from what Children Say and Do. In *Proceedings of the Childs Conception of Language*, pages 17–43. Springer Berlin Heidelberg.

Collobert, R. and Weston, J. (2008). A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In *Proceedings of the 25th International Conference on Machine Learning*, volume 20, pages 160–167.

Coucke, A., Saade, A., Ball, A., Bluche, T., Caulier, A., Leroy, D., Doumouro, C., Gisselbrecht, T., Caltagirone, F., Lavril, T., et al. (2018). Snips Voice Platform: an embedded Spoken Language Understanding system for private-by-design voice interfaces. *Computation and Language*.

Cowie, R. and Cornelius, R. R. (2003). Describing the Emotional States That Are Expressed in Speech. *Speech Communication*, 40(1-2):5–32.

Dai, A. M. and Le, Q. V. (2015). Semi-supervised Sequence Learning. In *Proceedings of the Advances in Neural Information Processing Systems*, volume 28, pages 3079–3087. Curran Associates, Inc.

Danescu-Niculescu-Mizil, C. and Lee, L. (2011). Chameleons in Imagined Conversations: A New Approach to Understanding Coordination of Linguistic Style in Dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 76–87. Association for Computational Linguistics.

Danescu-Niculescu-Mizil, C., Sudhof, M., Jurafsky, D., Leskovec, J., and Potts, C. (2013). A Computational Approach to Politeness with Application to Social

Factors. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 250–259. Association for Computational Linguistics.

Denil, M., Bazzani, L., Larochelle, H., and de Freitas, N. (2012). Learning Where to Attend with Deep Architectures for Image Tracking. *Neural Computation*, 24(8):2151–2184.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers forLanguage Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 4171–4186. Association for Computational Linguistics.

Ekman, P. (1992). An Argument for Basic Emotions. *Cognition & Emotion*, 6(3-4):169–200.

Ekman, P., Friesen, W. V., O'sullivan, M., Chan, A., Diacoyanni-Tarlatzis, I., Heider, K., Krause, R., LeCompte, W. A., Pitcairn, T., Ricci-Bitti, P. E., et al. (1987). Universals and Cultural Differences in the Judgments of Facial Expressions of Emotion. *Journal of Personality and Social Psychology*, 53(4):712–717.

El Hihi, S. and Bengio, Y. (1996). Hierarchical Recurrent Neural Networks for Long-Term Dependencies. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 493–499.

Elman, J. L. (1990). Finding Structure in Time. *Cognitive Science*, 14(2):179–211.

Fernández-Martínez, F., Zablotskaya, K., and Minker, W. (2012). Text categorization methods for automatic estimation of verbal intelligence. *Expert Systems with Applications*, 39(10):9807–9820.

Fernando, T., Denman, S., Sridharan, S., and Fookes, C. (2018). Soft + Hardwired Attention: An LSTM Framework for Human Trajectory Prediction and Abnormal Event Detection. *Neural networks*, 108:466–478.

Fischer, I. and Steiger, H.-J. (2020). Toward automatic evaluation of medical abstracts: The current value of sentiment analysis and machine learning for

classification of the importance of PubMed abstracts of randomized trials for stroke. *Journal of Stroke and Cerebrovascular Diseases*, 29(9):105042.

Fong, T., Nourbakhsh, I., and Dautenhahn, K. (2003a). A survey of socially interactive robots. *Robotics and Autonomous Systems*, 42(3-4):143–166.

Fong, T., Thorpe, C., and Baur, C. (2003b). Collaboration, Dialogue, and Human-Robot Interaction. *Proceedings of the 10th International Symposium of Robotics Research*, pages 255–266.

Fox, D. (2003). Adapting the Sample Size in Particle Filters Through KLD-Sampling. *The International Journal of Robotics Research*, 22(12):985–1003.

Gao, J., Galley, M., and Li, L. (2018). Neural Approaches to Conversational AI. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1371–1374. ACM.

Gardner, H. (2011). *Frames of Mind: The Theory of Multiple Intelligences*. Hachette UK.

Gibbs, R. W. and Van Orden, G. (2012). Pragmatic choice in conversation. *Topics in Cognitive Science*, 4(1):7–20.

Gladden, M. E. (2018). *Sapient Circuits and Digitalized Flesh: The Organization as Locus of Technological Posthumanization*. Defragmenter Media.

Globerson, A., Chechik, G., Pereira, F., and Tishby, N. (2007). Euclidean Embedding of Co-occurrence Data. *Journal of Machine Learning Research*, 8(Oct):2265–2295.

Glorot, X., Bordes, A., and Bengio, Y. (2011). Deep Sparse Rectifier Neural Networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, PMLR*, volume 15, pages 315–323. PMLR.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter Sentiment Classification using Distant Supervision. *CS224N Project Report, Stanford*, 1(12).

Godfrey, J. J., Holliman, E. C., and McDaniel, J. (1992). SWITCHBOARD: Telephone speech corpus for research and development. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages 517–520.

Grau, S., Sanchis, E., Castro, M. J., and Vilar, D. (2004). Dialogue act classification using a Bayesian approach. In *Proceedings of the 9th Conference Speech and Computer (SPECOM)*, pages 495–499.

Graves, A., Jaitly, N., and Mohamed, A. R. (2013). Hybrid speech recognition with Deep Bidirectional LSTM. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 273–278.

Graves, A., Liwicki, M., Fernández, S., Bertolami, R., Bunke, H., and Schmidhuber, J. (2008). A Novel Connectionist System for Unconstrained Handwriting Recognition. *IEEE transactions on pattern analysis and machine intelligence*, 31(5):855–868.

Grinberg, M. (2018). *Flask Web Development: Developing Web Applications with Python*. O'Reilly Media, Inc.

Grisettiyz, G., Stachniss, C., and Burgard, W. (2005). Improving Grid-based SLAM with Rao-Blackwellized Particle Filters by Adaptive Proposals and Selective Resampling. In *Proceedings of the International Conference on Robotics and Automation*, pages 2432–2437.

Grosz, B. J. (1982). Discourse Analysis. *Sublanguage. Studies of Language in Restricted Semantic Domains*, pages 138–174.

Gupta, U., Chatterjee, A., Srikanth, R., and Agrawal, P. (2017). A Sentiment-and-Semantics-Based Approach for Emotion Detection in Textual Conversations. *Proceedings of the Neu-IR 2017 SIGIR Workshop on Neural Information Retrieval*.

Hebb, D. O. (1949). *The Organization of Behavior: A Neuropsychological Theory*. Wiley.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. (2012). Improving Neural Networks by Preventing Co-adaptation of Feature Detectors. *Computing Research Repository (CoRR)*.

Hirschman, L. and Gaizauskas, R. (2001). Natural language question answering: the view from here. *Natural Language Engineering*, 7(4):275–300.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-term Memory. *Neural Computation*, 9(8):1735–1780.

Holmes, J. and Stubbe, M. (2015). *Power and Politeness in the Workplace: A Sociolinguistic Analysis of Talk at Work*. Routledge.

Hori, C. and Hori, T. (2017). End-to-end Conversation Modeling Track in DSTC6. *Workshop on Dialog System Technology Challenges (DSTC)*.

Hu, M. and Liu, B. (2004). Mining and Summarizing Customer Reviews. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 168–177.

Hubbard, D. J., Faso, D. J., Assmann, P. F., and Sasson, N. J. (2017). Production and perception of emotional prosody by adults with autism spectrum disorder. *Autism Research*, 10(12):1991–2001.

Hutto, C. J. and Gilbert, E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*, pages 216–225.

Ihasz, P. L. and Kryssanov, V. (2018). Emotions and Intentions Mediated with Dialogue Acts. In *Proceedings of the 5th International Conference on Business and Industrial Research (ICBIR)*, pages 125–130. IEEE.

Irsoy, O. and Cardie, C. (2014). Opinion Mining with Deep Recurrent Neural Networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 720–728.

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., and Hinton, G. E. (1991). Adaptive Mixtures of Local Experts. *Neural Computation*, 3(1):79–87.

Ji, Y., Haffari, G., and Eisenstein, J. (2016). A Latent Variable Recurrent Neural Network for Discourse Relation Language Models. In *Proceedings of the Conference North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 332–342. Association for Computational Linguistics.

Jordan, M. I. (1997). Serial Order: A Parallel Distributed Processing Approach. In *Advances in Psychology*, volume 121, pages 471–495. Elsevier.

Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., and Wu, Y. (2016). Exploring the Limits of Language Modeling. *Computation and Language*.

Jurafsky, D. (1997). Switchboard SWBD-DAMSL Shallow-Discourse-Function Annotation Coders Manual, draft 13. *Technical Report 97-01, University of Colorado Institute of Cognitive Science*, pages 225–233.

Jurafsky, D., Shriberg, E., and Biasca, D. (1997). Switchboard Dialog Act Corpus. Technical report, International Computer Science Inst. Berkeley CA.

Jurafsky, D., Shribergy, E., Fox, B., and Curl, T. (1998). Lexical, Prosodic, and Syntactic Cues for Dialog Acts. In *Proceedings of the ACL/COLING Workshop on Discourse Relations and Discourse Markers*. Association for Computational Linguistics.

Kadvany, J. (2016). Panini's Grammar and Modern Computation. *History and Philosophy of Logic*, 37(4):325–346.

Kalchbrenner, N. and Blunsom, P. (2013a). Recurrent Continuous Translation Models. In *Proceedings of the International Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1700–1709.

Kalchbrenner, N. and Blunsom, P. (2013b). Recurrent Convolutional Neural Networks for Discourse Compositionality. In *Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality*, pages 119–126. Association for Computational Linguistics.

Kalchbrenner, N., Espeholt, L., Simonyan, K., Oord, A. v. d., Graves, A., and Kavukcuoglu, K. (2016). Neural Machine Translation in Linear Time. *Computation and Language*.

Kamlasi, I. (2017). The Positive Politeness in Conversations Performed by the Students of English Study Program of Timor University. *Metathesis: Journal of English Language, Literature, and Teaching*, 1(2):68–81.

Kang, D. and Park, Y. (2014). Review-based measurement of customer satisfaction in mobile service: Sentiment analysis and VIKOR approach. *Expert Systems with Applications*, 41(4):1041–1050.

Kasper, G. (1990). Linguistic Politeness: Current research issues. *Journal of Pragmatics*, 14(2):193–218.

Kim, S. M. and Hovy, E. (2004). Determining the Sentiment of Opinions. In *Proceedings of the 20th International Conference on Computational Linguistics*, page 1367, Morristown, NJ, USA. Association for Computational Linguistics.

Kim, Y. (2014). Convolutional Neural Networks for Sentence Classification. *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751.

Kingma, D. and Ba, J. (2014). Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations*.

Krause, B., Lu, L., Murray, I., and Renals, S. (2016). Multiplicative LSTM for sequence modelling. *Workshop track of Proceedings of the International Conference on Learning Representations*.

Kumar, H., Agarwal, A., Dasgupta, R., Joshi, S., and Kumar, A. (2018). Dialogue Act Sequence Labeling using Hierarchical encoder with CRF. *AAAI Conference on Artificial Intelligence*, pages 3440–3447.

Lakomkin, E., Bothe, C., and Wermter, S. (2017). GradAscent at EmoInt-2017: Character and Word Level Recurrent Neural Network Models for Tweet Emotion Intensity Detection. In *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis at the Conference EMNLP*, pages 169–174. Association for Computational Linguistics.

Lakomkin, E., Zamani, M. A., Weber, C., Magg, S., and Wermter, S. (2018). EmoRL: Continuous Acoustic Emotion Classification using Deep Reinforcement Learning. In *International Conference on Robotics and Automation (ICRA)*, pages 4445–4450. IEEE.

Langlotz, A. and Locher, M. A. (2017). *(Im)politeness and Emotion*, pages 287–322. Palgrave Macmillan UK, London.

Larochelle, H. and Hinton, G. (2010). Learning to combine foveal glimpses with a third-order Boltzmann machine. In *Proceedings of the Conference on Advances in Neural Information Processing Systems*, pages 1243–1251. Curran Associates, Inc.

Latham, A. S. (1997). Learning Through Feedback. *Educational Leadership*, 54(8):86–87.

Lawrence, S., Giles, C. L., Tsoi, A. C., and Back, A. D. (1997). Face Recognition: A convolutional Neural-Network Approach. *IEEE Transactions on Neural Networks*, 8(1):98–113.

Lebret, R. and Collobert, R. (2013). Word Embeddings through Hellinger PCA. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 482–490. Association for Computational Linguistics.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lee, J. Y. and Dernoncourt, F. (2016). Sequential Short-Text Classification with Recurrent and Convolutional Neural Networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 515–520. Association for Computational Linguistics.

Lemaignan, S., Garcia, F., Jacq, A., and Dillenbourg, P. (2016). From Real-time Attention Assessment to "With-me-ness" in Human-Robot Interaction. In *Proceedings of the International Conference on Human Robot Interaction*, pages 157–164.

Levy, O. and Goldberg, Y. (2014). Neural Word Embedding as Implicit Matrix Factorization. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 2177–2185.

Li, X. and Wu, X. (2015). Constructing Long Short-Term Memory based Deep Recurrent Neural Networks for Large Vocabulary Speech Recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4520–4524.

Liu, Y., Han, K., Tan, Z., and Lei, Y. (2017). Using Context Information for Dialog Act Classification in DNN Framework. In *Proceedings of the Conference*

on Empirical Methods in Natural Language Processing (EMNLP), pages 2160–2168. Association for Computational Linguistics.

Loper, E. and Bird, S. (2002). NLTK: the Natural Language Toolkit. *Proceedings of the ACL-2 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, 1:63–70.

López-Cózar, R., Silovsky, J., and Griol, D. (2010). F2 - New Technique for Recognition of User Emotional States in Spoken Dialogue Systems. In *Proceedings of the 11th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 281–288. Association for Computational Linguistics.

Lu, Y., Srivastava, M., Kramer, J., Elfardy, H., Kahn, A., Wang, S., and Bhardwaj, V. (2019). Goal-Oriented End-to-End Conversational Models with Profile Features in a Real-World Setting. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Industry Papers)*, pages 48–55, Minneapolis - Minnesota. Association for Computational Linguistics.

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., and Potts, C. (2011). Learning Word Vectors for Sentiment Analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 142–150. Association for Computational Linguistics.

Maaten, L. v. d. and Hinton, G. (2008). Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605.

Macherey, K., Och, F. J., and Ney, H. (2001). Natural Language Understanding Using Statistical Machine Translation. In *Proceedings of the Seventh European Conference on Speech Communication and Technology*.

MacWhinney, B. (1991). *The CHILDES project: Tools for analyzing talk.* Lawrence Erlbaum Associates Publishers.

Mavridis, N. (2015). A review of verbal and non-verbal human–robot interactive communication. *Robotics and Autonomous Systems*, 63:22–35.

McCrae, P. (2010). *A Computational Model for the Influence of Cross-Modal Context upon Syntactic Parsing.* PhD thesis, University of Hamburg.

McHugh, M. L. (2012). Interrater reliability: the kappa statistic. *Biochemia Medica*, 22(3):276–282.

McKeown, G., Valstar, M., Cowie, R., Pantic, M., and Schroder, M. (2012). The SEMAINE Database: Annotated Multimodal Records of Emotionally Colored Conversations between a Person and a Limited Agent. *IEEE Transactions on Affective Computing*, 3(1):5–17.

McTear, M., Callejas, Z., and Griol, D. (2016). *The Conversational Interface: Talking to Smart Devices*. Springer.

McTear, M. F. (2002). Spoken Dialogue Technology: Enabling the Conversational User Interface. *ACM Computing Surveys (CSUR)*, 34(1):90–169.

Meng, Z., Mou, L., and Jin, Z. (2017). Hierarchical RNN with Static Sentence-Level Attention for Text-Based Speaker Change Detection. In *Proceedings of the ACM Conference on Information and Knowledge Management*, pages 2203–2206.

Mesnil, G., Dauphin, Y., Yao, K., Bengio, Y., Deng, L., Hakkani-Tur, D., He, X., Heck, L., Tur, G., Yu, D., et al. (2015). Using Recurrent Neural Networks for Slot Filling in Spoken Language Understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 23(3):530–539.

Mesnil, G., He, X., Deng, L., and Bengio, Y. (2013). Investigation of Recurrent-Neural-Network Architectures and Learning Methods for Spoken Language Understanding. In *Proceedings of the Interspeech*, pages 3771–3775. International Speech Communication Association (ISCA).

Mikolov, T., Corrado, G., Chen, K., and Dean, J. (2013a). Efficient Estimation of Word Representations in Vector Space. *Proceedings of the International Conference on Learning Representations (ICLR 2013)*, pages 1–12.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed Representations of Words and Phrases and their Compositionality. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 3111–3119.

Misra, D. K., Sung, J., Lee, K., and Saxena, A. (2016). Tell Me Dave: Context-Sensitive Grounding of Natural Language to Manipulation Instructions. *The International Journal of Robotics Research*, 35(1-3):281–300.

Mohammad, S. M. and Bravo-Marquez, F. (2017a). Emotion Intensities in Tweets. In *Proceedings of the Sixth Joint Conference on Lexical and Computational Semantics (*Sem)*, Vancouver, Canada.

Mohammad, S. M. and Bravo-Marquez, F. (2017b). WASSA-2017 shared task on emotion intensity. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA)*, Copenhagen, Denmark.

Mundra, S., Sen, A., Sinha, M., Mannarswamy, S., Dandapat, S., and Roy, S. (2017). Fine-Grained Emotion Detection in Contact Center Chat Utterances. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 337–349. Springer.

Neviarouskaya, A., Prendinger, H., and Ishizuka, M. (2007). Textual Affect Sensing for Sociable and Expressive Online Communication. *Affective Computing and Intelligent Interaction*, pages 218–229.

Nijdam, N. A. (2009). *Mapping Emotion to Color*. Citeseer.

O'Dea, B., Wan, S., Batterham, P. J., Calear, A. L., Paris, C., and Christensen, H. (2015). Detecting suicidality on Twitter. *Internet Interventions*, 2(2):183–188.

Ortega, D. and Vu, N. T. (2017). Neural-based Context Representation Learning for Dialog Act Classification. In *Proceedings of the Conference of the Special Interest Group on Discourse and Dialogue*, pages 247–252.

Osgood, C. E., May, W. H., Miron, M. S., and Miron, M. S. (1975). *Cross-cultural Universals of Affective Meaning*, volume 1. University of Illinois Press.

O'Shea, J., Bandar, Z., and Crockett, K. (2012). A Multi-classifier Approach to Dialogue Act Classification Using Function Words. In *Transactions on Computational Collective Intelligence VII*, pages 119–143. Springer.

Pandey, A. and Gelin, R. (2018). A Mass-Produced Sociable Humanoid Robot: Pepper: The First Machine of Its Kind. *IEEE Robotics Automation Magazine*, pages 40–48.

Pandey, A. K. (2012). *Towards Socially Intelligent Robots in Human Centered Environment*. PhD thesis, Institut National des Sciences Appliquées de Toulouse (INSA).

Pang, B. and Lee, L. (2008). Opinion Mining and Sentiment Analysis. *Foundations and Trends in Information Retrieval*, 2(12):1–135.

Pennington, J., Socher, R., and Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. *Proceedings of the Conference on EMNLP*, pages 1532–1543.

Perera, V., Pereira, T., Connell, J., and Veloso, M. (2017). Setting Up Pepper For Autonomous Navigation And Personalized Interaction With Users. *Robotics*.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep Contextualized Word Representations. *North American Chapter of the Association for Computational Linguistics (NAACL 2018)*, pages 2227–2237.

Poria, S., Hazarika, D., Majumder, N., Naik, G., Cambria, E., and Mihalcea, R. (2019). MELD: A Multimodal Multi-Party Dataset for Emotion Recognition in Conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 527–536. Association for Computational Linguistics.

Poria, S., Majumder, N., Mihalcea, R., and Hovy, E. (2019). Emotion Recognition in Conversation: Research Challenges, Datasets, and Recent Advances. *IEEE Access*, 7:100943–100953.

Radford, A., Jozefowicz, R., and Sutskever, I. (2017). Learning to Generate Reviews and Discovering Sentiment. *Computation and Language*.

Rajagopalan, K. (2000). On Searle [on Austin] on language. *Language & Communication*, 20(4):347–391.

Rashkin, H., Smith, E. M., Li, M., and Boureau, Y.-L. (2018). I Know the Feeling: Learning to Converse with Empathy. *Computation and Language*.

Rau, P.-L. P., Li, Y., and Liu, J. (2013). Effects of a Social Robot's Autonomy and Group Orientation on Human Decision-Making. *Advances in Human-Computer Interaction*.

Riggio, R. E. (1986). Assessment of basic social skills. *Journal of Personality and social Psychology*, 51(3):649.

Ritter, A., Cherry, C., and Dolan, W. B. (2011). Data-Driven Response Generation in Social Media. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 583–593. Association for Computational Linguistics.

Rogers, P. S. and Lee-Wong, S. M. (2003). Reconceptualizing Politeness to Accommodate Dynamic Tensions in Subordinate-to-Superior Reporting. *Journal of Business and Technical Communication*, 17(4):379–412.

Russell, J. A. and Mehrabian, A. (1977). Evidence for a Three-Factor Theory of Emotions. *Journal of Research in Personality*, 11(3):273–294.

Sailunaz, K., Dhaliwal, M., Rokne, J., and Alhajj, R. (2018). Emotion Detection from Text and Speech - A Survey. *Social Network Analysis and Mining*, 8(1):28.

Salem, M., Ziadee, M., and Sakr, M. (2014). Marhaba, how may I help you?: Effects of Politeness and Culture on Robot Acceptance and Anthropomorphization. In *Proceedings of the International Conference on Human-robot Interaction*, pages 74–81.

Salovey, P. and Mayer, J. D. (1990). Emotional Intelligence. *Imagination, Cognition and Personality*, 9(3):185–211.

Saville-Troike, M. (2008). *The Ethnography of Communication*, volume 14. John Wiley & Sons.

Sbisà, M. (2002). Speech acts in context. *Language & Communication*, 22(4):421–436.

Schuster, M. and Paliwal, K. K. (1997). Bidirectional Recurrent Neural Networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681.

Searle, J. R. (1979). *Expression and Meaning: Studies in the Theory of Speech Acts*. Cambridge University Press.

Searle, J. R. and Searle, J. R. (1969). *Speech acts: An essay in the philosophy of language*. Cambridge University Press.

Serban, I. V., Lowe, R., Henderson, P., Charlin, L., and Pineau, J. (2015). A Survey of Available Corpora For Building Data-Driven Dialogue Systems: The Journal Version. *Dialogue & Discourse*, 9(1):1–49.

Sheridan, T. B. and Verplank, W. L. (1978). Human and Computer Control for Undersea Teleoperators. Technical report, Massachusetts Inst of Tech Cambridge Man-Machine Systems Lab.

Shi, W. and Yu, Z. (2018). Sentiment Adaptive End-to-End Dialog Systems. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1509–1519. Association for Computational Linguistics.

Shriberg, E., Dhillon, R., Bhagat, S., Ang, J., and Carvey, H. (2004). The ICSI Meeting Recorder Dialog Act (MRDA) Corpus. Technical report, International Computer Science Inst. Berkeley CA.

Shriberg, E., Stolcke, A., Jurafsky, D., Coccaro, N., Meteer, M., Bates, R., Taylor, P., Ries, K., Martin, R., and Van Ess-Dykema, C. (1998). Can Prosody Aid the Automatic Classification of Dialog Acts in Conversational Speech? *Language and Speech*, 41(3-4):443–492.

Skantze, G. (2007). Error Handling in Spoken Dialogue Systems-Managing Uncertainty, Grounding and Miscommunication: Chapter 2, Spoken Dialogue Systems. *KTH Computer Science and Communication*.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. (2013a). Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642.

Socher, R., Perelygin, A., Wu, J. Y., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. (2013b). Recursive Deep Models for Semantic Compositionality over a Sentiment Treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1631–1642. Association for Computational Linguistics.

Sordoni, A., Galley, M., Auli, M., Brockett, C., Ji, Y., Mitchell, M., Nie, J.-Y., Gao, J., and Dolan, B. (2015). A Neural Network Approach to Context-

Sensitive Generation of Conversational Responses. *Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL*, pages 196–205.

Speer, R., Chin, J., and Havasi, C. (2017). ConceptNet 5.5: An Open Multilingual Graph of General Knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4444–4451.

Srinivasan, V. and Takayama, L. (2016). Help Me Please: Robot Politeness Strategies for Soliciting Help From People. In *Proceedings of the Conference on Human Factors in Computing Systems*, pages 4945–4955. ACM.

Steinfeld, A., Fong, T., Kaber, D., Lewis, M., Scholtz, J., Schultz, A., and Goodrich, M. (2006). Common Metrics for Human-Robot Interaction. In *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, pages 33–40. ACM.

Stolcke, A., Ries, K., Coccaro, N., Shriberg, E., Bates, R., Jurafsky, D., Taylor, P., Martin, R., Van Ess-Dykema, C., and Meteer, M. (2000). Dialogue Act Modeling for Automatic Tagging and Recognition of Conversational Speech. *Computational Linguistics*, 26(3):339–373.

Suddrey, G., Jacobson, A., and Ward, B. (2018). Enabling a Pepper Robot to provide Automated and Interactive Tours of a Robotics Laboratory. *Robotics*.

Surendran, D. and Levow, G.-A. (2006). Dialog act tagging with support vector machines and hidden Markov models. In *Interspeech – ICSLP*, pages 1950–1953.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to Sequence Learning with Neural Networks. In *Proceedings of the Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Tadele, T. S., de Vries, T., and Stramigioli, S. (2014). The Safety of Domestic Robotics: A Survey of Various Safety-Related Publications. *IEEE Robotics & Automation Magazine*, 21(3):134–142.

Tavafi, M., Mehdad, Y., Joty, S. R., Carenini, G., and Ng, R. T. (2013). Dialogue Act Recognition in Synchronous and Asynchronous Conversations. In

Proceedings of the Conference of the Special Interest Group on Discourse and Dialogue, pages 117–121. Association for Computational Linguistics.

Thelwall, M., Buckley, K., and Paltoglou, G. (2012). Sentiment Strength Detection for the Social Web. *JASIST*, 63(1):163–173.

Tran, Q. H., Zukerman, I., and Haffari, G. (2017). Preserving Distributional Information in Dialogue Act Classification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2141–2146. Association for Computational Linguistics.

Ultes, S., Rojas Barahona, L. M., Su, P.-H., Vandyke, D., Kim, D., Casanueva, I., Budzianowski, P., Mrkšić, N., Wen, T.-H., Gasic, M., and Young, S. (2017). PyDial: A Multi-domain Statistical Dialogue System Toolkit. In *Proceedings of the Association for Computational Linguistics 2017, System Demonstrations*, pages 73–78. Association for Computational Linguistics.

van Gerven, M. and Bohte, S. (2018). Editorial: Artificial Neural Networks as Models of Neural Information Processing. *Artificial Neural Networks as Models of Neural Information Processing, Frontiers Media SA*, page 5.

Van Harmelen, F., Lifschitz, V., and Porter, B. (2008). *Handbook of Knowledge Representation*, volume 1. Elsevier.

Vanzo, A., Croce, D., and Basili, R. (2014). A context-based model for Sentiment Analysis in Twitter. In *Proceedings of the International Conference on Computational Linguistics: Technical Papers*, pages 2345–2354.

Vinyals, O., Kaiser, Ł., Koo, T., Petrov, S., Sutskever, I., and Hinton, G. (2015). Grammar as a Foreign Language. In *Proceedings of the Conference on Advances in Neural Information Processing Systems*, pages 2773–2781.

Vinyals, O. and Le, Q. (2015). A Neural Conversational Model. In *Proceedings of the International Conference on Machine Learning: Deep Learning Workshop*.

Wang, S. and Manning, C. D. (2012). Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 90–94. Association for Computational Linguistics.

Watts, R. J. (2003). *Politeness*. Cambridge University Press.

Wermter, S. (1995). *Hybrid Connectionist Natural Language Processing*, volume 7. Chapman & Hall, London.

Wermter, S. and Löchel, M. (1996). Learning dialog act processing. In *Proceedings of the 16th Conference on Computational Linguistics*, volume 2, pages 740–745. Association for Computational Linguistics.

Wermter, S. and Sun, R. (2000). *Hybrid Neural Systems*. Number 1778. Springer Science & Business Media.

Weston, J. (2016). Dialog-based Language Learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Proceedings of the Advances in Neural Information Processing Systems (NIPS) 29*, pages 829–837. Curran Associates, Inc.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In *Proceedings of the HLT/EMNLP 2005, Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 347–354.

Wood, I., McCrae, J. P., Andryushechkin, V., and Buitelaar, P. (2018). A Comparison Of Emotion Annotation Schemes And A New Annotated Data Set. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018*, pages 1197–1202.

Wooffitt, R. (2005). *Conversation Analysis and Discourse Analysis: A Comparative and Critical Introduction*. Sage.

Yadav, S., Ekbal, A., Saha, S., and Bhattacharyya, P. (2018). Medical Sentiment Analysis using Social Media: Towards building a Patient Assisted System. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018*, pages 2790–2797.

Yang, X., Chen, Y.-N., Hakkani-Tür, D., Crook, P., Li, X., Gao, J., and Deng, L. (2017). End-to-End Joint Learning of Natural Language Understanding and Dialogue Manager. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5690–5694.

Yang, X., Liu, J., Chen, Z., and Wu, W. (2014). Semi-supervised Learning of Dialogue Acts Using Sentence Similarity Based on Word Embeddings. In *Proceedings of International Conference on Audio, Language and Image Processing*, pages 882–886.

Zamani, M. A., Magg, S., Weber, C., Wermter, S., and Fu, D. (2018). Deep Reinforcement Learning using Compositional Representations for Performing Instructions. *Paladyn, Journal of Behavioral Robotics*, 9(1):358–373.

Zhang, X., Zhao, J., and LeCun, Y. (2015). Character-level Convolutional Networks for Text Classification. In *Proceedings of the Advances in Neural Information Processing Systems*, pages 649–657.

Zhao, B., Li, X., and Lu, X. (2017). Hierarchical recurrent neural network for video summarization. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 863–871.

Zhou, P., Shi, W., Tian, J., Qi, Z., Li, B., Hao, H., and Xu, B. (2016). Attention-based Bidirectional Long Short-term Memory Networks for Relation Classification. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, volume 2, pages 207–212. Association for Computational Linguistics.

Declaration on Oath

Eidesstattliche Versicherung

I hereby declare, on oath, that I have written the present dissertation by my own and have not used other than the acknowledged resources and aids.

Hiermit erkläre ich an Eides statt, dass ich die vorliegende Dissertationsschrift selbst verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

Hamburg, 18.08.2020 Chandrakant Bothe

City, Date Signature
Ort, Datum *Unterschrift*

Declaration on Publication

Erklärung zur Veröffentlichung

I agree to the placement of the dissertation in the library of the Department of Computer Science.
Ich stimme der Einstellung der Dissertation in die Bibliothek des Fachbereichs Informatik zu.

Hamburg, 25.11.2020 Chandrakant Bothe

City, Date Signature
Ort, Datum *Unterschrift*