Modeling Thematic Changes in a Collaborative Task-Oriented Dialogue

Eliza Margaretha

Master Thesis

Thesis Committee:
Prof. Dr. Stephan Busemann (Saarland University)
Dr. Ir. Geert-Jan Kruijff (DFKI)
Dr. Gosse Bouma (University of Groningen)

European Masters Program in
Language and Communication Technologies (LCT)

Computational Linguistics
Saarland University

Faculty of Arts
University of Groningen

Saarbrücken, Germany
October 2011

I hereby confirm that the thesis presented here is my own work, with all assistance acknowledged.

Saarbrücken, 28 October 2011

Eliza Margaretha

Abstract

When people talk, they focus their attention to the theme of their conversation. As a dialogue progresses, they change their focus of attention and the dialogue theme accordingly. In this work, we investigate the mechanism behind the changes of dialogue themes (i.e. thematic changes) in a collaborative task-oriented dialogue. We introduce an approach to building a thematic structure which can describe these changes and predict the next dialogue theme. A thematic structure of this kind can be useful for various tasks such as for improving speech recognition outputs or for helping to recognize the intention behind a user's utterance. We aim at building a thematic structure which can also model thematic changes in human-robot interactions.

Our approach uses a Markov Logic Network (MLN) as a probabilistic logic model. For this model, we specify logical rules to characterize the mechanisms of thematic changes. We train and test a number of MLN models using a human-human dialogue data set which is considered similar to human-computer dialogue data in USAR. The MLN models are compared to two kinds of baselines, namely a random and an informed baseline. The random baseline is a random choice over all possible dialogue theme, and the informed baseline is a basic focus prediction model which always continues the dialogue themes of the previous utterance. The experiment results show that the MLN models outperform both baselines. Furthermore, we argue that our approach can be used for structuring the dialogue themes in a human-robot collaborated dialogue in performing an urban search-and-rescue (USAR) task.

Acknowledgement

I would like to express my gratitude to my reviewers, Prof. Stephan Busemann, Dr. Geert-Jan Kruijff and Dr. Gosse Bouma for supervising me in writing this thesis and for their invaluable advice. The work in this thesis could not have been accomplished without my supervisor at DFKI (German Research Center for Artificial Intelligence), Miroslav Janíček. I wish to thank him for his guidance, support and patience in helping me to develop this work. Moreover, I would like to acknowledge Dr. Ivana Kruijff-Korbajova for her comments.

I thank the Erasmus Mundus LCT consortium for giving me the opportunity to pursue my master degree and the DFKI for supporting this research. Thanks for all support from Maria Nadejde, Prashanth Rao, Kak Heru Surya-Atmadja, Martin Leonard Tangel, Ronny, and Tri Kurniawan Wijaya. Especially, I am grateful to Jelke Bloem for helping me improving this thesis, and I am obliged to Jens Illig for a lot of help, suggestions and encouragement. Finally, I am thankful to my family for their constant love and support.

Contents

Chapter 1

Introduction

When people talk, they always talk about some topic, which is the theme of their conversation. As their conversation progresses, they may continue talking about the same theme or talk more about its details. Besides, they may change the theme of their conversation. An utterance of a speaker always expresses his/her intentions in saying what he/she says. In a collaborative task-oriented dialogue, the intentions can be what a speaker intends to do, or what he/she intends his/her hearer to do, in order to accomplish the goal of the task.

The work in this thesis investigates how the theme of a conversation changes, specifically in a collaborative task-oriented dialogue. We attempt to model a thematic structure for human-robot interactions in performing an Urban Search and Rescue (USAR) task. A thematic structure is a structure of the themes of dialogue segments in a dialogue. We are interested in using an intentional approach to modeling a thematic structure.

The following sections of this chapter give an overview of our work. We begin with a description of the motivation behind the work. Subsequently, we describe the problem that is to be solved and the objectives of the work in sections 1.2 and 1.3, respectively. In section 1.5, we highlights the contributions of the work in this thesis. Finally, we end this chapter with an outline of the thesis content in section 1.6.

1.1 Motivation

A conversation between two or more people always involves a joint activity [Clark, 1996]. Two people converse to arrange an appointment, to solve a problem or to perform a task. These people are engaged in one shared activity and they both contribute to undertaking it. A joint activity usually comprises a series of joint actions i.e. actions which the participants perform in coordination with each other. In performing a joint activity, a participant may disagree with the intention of another participant or they may negotiate to come up with the best plan to achieve their joint-goal.

The conversation in which two or more people coordinate with each other in order to accomplish a shared task, is designated as a collaborative task-oriented dialogue. The short conversation (1.1.1) below is a collaborative task-oriented dialogue wherein Bill and Ann coordinate with each other to set a meeting time. They collaborate to attain a common goal - arranging a meeting time which works for both of them. This activity splits into two joint actions. Firstly, they arrange the meeting day. Bill makes a suggestion to meet on Friday and Ann simply accepts it. Secondly, Ann indicates her available time on Friday afternoon and Bill proposes a specific time. Ann implicitly disapproves Bill's proposal and suggests another time. The action ends with Bill agreeing on the suggested time and giving Ann an explicit acknowledgement.

(1.1.1) Bill: Do you have time on Friday?
Ann: Yes, I will have some time in the afternoon.
Bill: At 2 pm?
Ann: How about 3?
Bill: Ok

What the participants intend to do, accounts for their joint action, and this intention establishes a dialogue theme, i.e the main or the most salient entity being talked about in a dialogue at a given point. The concept of dialogue theme is similar to that of the focus of attention [Grosz and Sidner, 1986], discourse focus [Sidner, 1979], global theme or global topic [Hirst, 1981]. The intention of a speaker is a pragmatic phenomenon because it depends on the context of a conversation. In the context of the dialogue (1.1.1), Bob has an intention to meet Ann on Friday. For success in a collaborative communication, mutual understanding about what is being talked is necessary, and both participants would have to agree on it. By asking "At 2 pm?", Bob implicitly agrees on Ann's intention to meet in the afternoon. Thus, they both now focus their attention on the same entity "some time in the afternoon" and it becomes the dialogue theme.

Note that, the level of a dialogue theme used in a system is determined by the purpose of interpreting a dialogue. The whole dialogue (1.1.1) has a general theme of "setting a meeting time". This theme can be broken down into more local dialogue themes (e.g. "appointment date"). An appointment scheduling system, such as VERBMOBIL [Wahlster, 2000] and COSMA [Busemann et al., 1994], requires the more specific dialogue themes. Since it aims to schedule specific date and time for an appointment, it deals with the dialogue themes "appointment date" and "appointment time".

Having a common focus of attention ensures the participants' mutual understanding and enables them to perform a collaborative joint activity. Common ground, mutually recognized shared information or beliefs [Stalnaker, 2002], is a prerequisite providing the context of a conversation. Only with mutual belief, the participants believe that they are talking about the same thing and are taking part in the same joint action. Grounding (i.e. the process of creating, adding or modifying common ground) is an incremental process. As a conversation progresses, dialogue participants add new information to their common ground or modify the existing common ground. Also, the dialogue participants' focus of attention changes according to their common ground. In the sample dialogue (1.1.1), the theme of the conversation changes from "time on Friday" to "some time in the afternoon". In this example, the dialogue participants shift their current focus of attention to a more specific property of the current dialogue theme (i.e. the next dialogue theme). Therefore, there is a zooming-in process in this thematic change.

Like human-human dialogues, human-computer dialogues contain dialogue themes and they also change as the dialogues progress. Thus, a user and a computer agent also need a mutual understanding about the theme of their conversation at each point of time. With mutual understanding and common ground, they can then collaborate in a joint activity. To establish the common ground, a computer agent should be able to recognize what a user is talking about. When a user talks about a specific object mentioned before, it should be able to recognize what the user refers to. A computer agent should also be able to understand what a user would like to do, and what a user wants from it. Additionally, the agent should give response about what the user has said. It should be able to talk about dialogue themes. Thus, the conversation between the user and the agent can keep going on, and they can achieve a joint-goal in a joint-activity. Depending on the level of its autonomy, the agent might also be able to suggest a plan and perform automatic movements without explicit instructions by its user.

In practice, a dialogue data is often noisy. By noisy, we means the speech may not be clear enough due to the background noise in a noisy environment. Particularly, a dialogue in a USAR setting, is very likely to be noisy because an accident site is not an isolated room. A damaged apparatus such as a gas pipe may produce loud sounds disturbing the conversation of a rescue team. Noisy speech is difficult to recognize and a dialogue system usually tries to recognize what a user said without any clue regarding the theme of the talk. As a results, a speech recognizer tends to produce a lot of errors and irrelevant words in its outputs. Moreover, in a dialogue system, the speech recognition errors are likely to be exacerbated in other modules following the automatic speech recognition (ASR) module.

A thematic structure which is also able to predict what a user will be likely to talk about, will benefit an ASR module. By predicting the themes which are likely to appear in the next user utterance, it can expect what a user will say. The predictions (i.e. predicted themes) can be used provide a context to prime ASR outputs, especially in recognizing a noisy user utterance. Thus, the speech recognition errors can be reduced by using this context [Lison, 2008]. Moreover, predicted themes can be used to help to recognize the intention of a speaker. For these reasons, in this work, we would like to build a thematic structure which does not only structures thematic changes, but can also predict what the next theme could be.

1.2 Problem

The problem we attempt to solve in this work is the problem of structuring the changes of a dialogue theme in a collaborative task-oriented dialogue. Moreover, we try to discover how to predict such a change, which we call a thematic change. In this work, we specifically observe three types of thematic changes:

1. *Continuation*: a continuation of a dialogue theme of the last utterance to the next utterance
2. *Shift New*: an appearance of a new theme in the next utterance
3. *Shift Old*: a re-appearance of a recent theme which has been talked about before, but not in the last utterance

We are interested in building a thematic structure that is able to predict the next theme or what a user is likely to say next.

Existing approaches can be used to structure thematic changes to some extent. However, some approaches can only describes thematic changes in a dialogue and do not predict the next theme. Other approaches can be used to predict thematic changes, but they can not predict all types of thematic changes. Some approaches are not robust because they rely too much on a well-structured representation of dialogue segments and their relations. On the other hand, some other approaches do not accomodate a rich thematic structure.

Grosz and Sidner [1986] have proposed a stack mechanism to structure thematic changes based on the dominance hierarchy of dialogue segment purposes or speaker intentions. We argue that the stack mechanism is rather not flexible, because the themes in the stack are not easily accessible without removing all the themes on top of it. Therefore, it may not be suitable for a dialogue where the dialogue participants change the themes frequently and not according to the order in the stack.

The problem of structuring thematic changes can also be seen as structuring the salience of dialogue segments. The term salience refers to the things which are the most important or relevant compared to other things in a given context of information [Chiarcos et al., 2011]. The salience of a dialogue segment is very similar to the theme of a dialogue segment, because it also refers to the central thing being talked about in a dialogue segment. Grosz et al. [1995] introduced a framework to model the salience between discourse segments by using centering theory.

Centering theory can model the continuation relationship between the dialogue themes in two neighboring utterances. However, people do not always continue to talk about a theme. They may also talk about a new theme or a theme older than the one of the last dialogue segment. The centering theory models this as a shift transition, however, it can neither specifically predict an appearance of a new theme nor can it connect the themes of two dialogue segments which are not adjacent. Prince [1992] explained information status which can be used to identify whether a theme in a dialogue segment is new or old. However, she did not describe how to predict whether there will be a new or an old theme in the next dialogue segment.

The problem of predicting the next theme is rather different from that of identifying a theme. When predicting the next theme, we are not informed about the next utterance. When indentifying a theme, we observe the information in an utterance and take a benefit out of it to identify the theme expressed in the utterance. Anaphora resolution can be seen as an analog of a theme identification problem. A theme can be considered as an anaphoric or referring expression, which refers to a theme in a previous utterance as its antecedent or referent. By doing anaphora resolution, we can identify a theme and eventually structure thematic changes. Nevertheless, anaphora resolution requires knowledge about an anaphor to resolve in the next utterance. Thus, it does not predict the next theme.

Segmented Discourse Representation Theory (SDRT) introduced by Lascarides and Asher [2007] can be used for anaphora resolution and for describing the thematic changes between discourse segments. However, SDRT works for systems which require highly structured discourse context. Thus, it seems to be too complicated for other systems which only use simple structures. Moreover, noisy dialogue data will be difficult to model using SDRT due to its complex structure. Therefore, SDRT may not be robust enough to model thematic changes in noisy dialogue data.

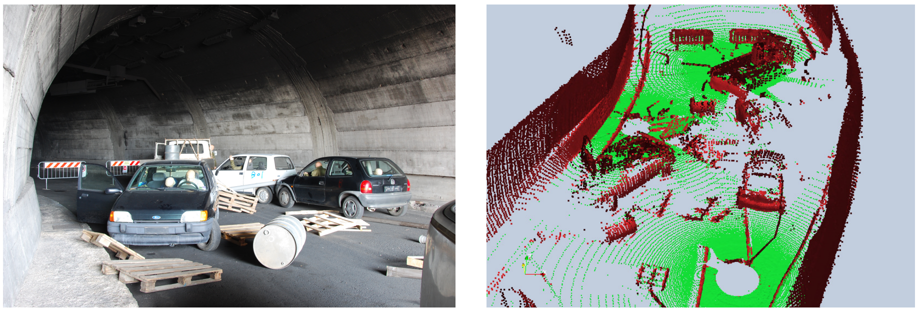
Furthermore, thematic changes can be structured by using a knowledge-based approach. A knowledge base may contain an ontology of activities, that is a set of concepts describing how activities are performed step by step in certain ways. An advantage of using a knowledge-based approach is that it can explain why a dialogue develops towards a certain direction. Using a knowledge base, we can predict how a dialogue may progress, and therefore we can also predict thematic changes. This approach, however, suffers from several limitations. For instance, a knowledge-based approach is rather domain specific because an ontology is dedicated to be used in a particular domain. Thus, a knowledge-based approach requires different ontologies to structure thematic changes of dialogues from different domains. Moreover, an ontology is not easy to build and maintain, especially if it needs to be scaled up to many or to broad domains. An ontology can represent a simple and well-structured dialogue well. However, it may not be neat to represent a free-ranging dialogue or a dynamic dialogue. The thematic changes in a dynamic dialogue do not follow a certain order. Instead, dialogue themes can appear and re-appear in any order. An ontology for such a dialogue would have a very complex structure.

Since the existing approaches still suffer from various drawbacks, we try to come up with a new approach to structure thematic changes. We also attempt to structure dialogue themes in human-robot interactions in a situated dialogue, particularly in a USAR setting [Murphy et al., 2008]. A situated dialogue is a dialogue happening at a certain time and place.

1.3 Objective

The objective of this thesis is to develop a dialogue structure which models thematic changes in a collaborative and task-oriented dialogue. In such a dialogue, the participants work together to accomplish a well-defined task as their shared-goal. Firstly, we would like to observe the aspects that account for thematic changes in a task-oriented dialogue. Secondly, we would like to observe how intentional aspects can be useful for predicting thematic changes. In this work, we try to build a thematic structure which is able to predict the three types of thematic changes described in the previous section. Our approach based on the intentional perspective on using language, therefore it works at the intentional level rather than at the text level.

Furthermore, we attempt to build a thematic structure for human-robot interactions in performing a USAR task. A USAR task typically consists of well-structured strategies and many sub-tasks. For instance, in firefighting, firefighters have a checklist of tasks they should to do in a certain order [Klaene and Sanders, 2008]. In carrying a USAR task, an off-site operator can work remotely with a robot to explore a disaster area and locate victims. A rescue robot plays a role as a member of a rescue team and gives support to the team. It is typically useful for first exploring a disaster site which is probably still dangerous for human rescuers to approach - for example, because of smoke and high air temperature.



(a) (b)

Figure 1.3.1: (a) Tunnel accident setup at the *Scuola di Formazione Operativa* (SFO) in Montelibretti, Italy (b) Map of the setup

Figure 1.3.1 shows a tunnel accident setting which was built to investigate how human rescuers work in reality and communicate with each other in a performing a USAR task. In this setting, a lorry drove into a tunnel and lost its load of barrels and pallets which caused an accident involving multiple cars. A group of firefighters was employed to explore the accident and their conversations were recorded. The transcripts show that the firefighters change their focus of attention frequently during the exploration. Their focus of attention changes in accordance with how the dialogue progresses at each point of time. The focus of attention does not always refer to the most recent things being talked about, but may also refer to something being talked about some time ago. In this work, we attempt to build a thematic structure which reflects such dynamic thematic/focus changes.

Since a rescue robot plays a role as an assistant in a rescue team, it is supposed to collaborate in a rescue mission. By predicting what the most plausible next dialogue themes are, a rescue robot would be able to better understand the conversation in a rescue team. Moreover, a rescue robot should not only be able to determine what is being talked about in the team, but it may also suggest a plan for accomplishing a rescue mission. For these reasons, a dynamic thematic structure which can predict the next theme is necessary for a dialogue system in this domain.

1.4 Evaluation

Evaluation of a thematic structure in a dialogue system is complex, because a dialogue system itself is complex. A dialogue system consists of various modules, such as automatic speech recognition, natural language understanding, dialogue management and natural language generation modules. A thematic structure module may require inputs from other modules and provide its outputs as inputs to other modules. In practice, the evaluation of a thematic structure module should involve the performance of the modules related to it. The evaluation of a thematic structure in a human-robot dialogue system depends on the performance of many modules including the mobility of the robot, the level of automation, and so on.

Particularly in a situated dialogue, the evaluation may be highly influenced by the performance or the results of an image recognition module. This is because a situated dialogue is influenced by spatial and temporal aspects. In a situated dialogue, the dialogue theme at some point of time always depends on the situation of the dialogue participants at that time. For instance, the participants may talk about the specific things they see or hear in their current location during the conversation. Therefore, in human-robot interactions, a robot is required to be able to visually recognize its environment and objects in the environment.

Ideally, evaluating a thematic structure requires measuring the performance of the whole system. In a development stage, however, some of modules of the system may have not been built yet. Therefore, we cannot really measure the whole system's performance. Alternatively, we can simplify the evaluation by performing unit or module testing, instead of testing the whole system. In this work, we evaluate the thematic structure as a module of a system regardless of other modules in the system.

Moreover, evaluating predictions of the theme in the next utterance is not simple, because a successful prediction is not always easy to define. In an identical situation, two people may give different responses expressing different dialogue themes. This suggests that there are multiple options for dialogue themes at some point in time. Although a single theme can be the most suitable for one to talk about, other themes may still be acceptable. An evaluation against a gold standard, therefore, does not necessarily show an exact accuracy, but only a result measured relative to the consulted annotation.

A thematic structure in a human-robot dialogue system can be evaluated by using a large collection of data. However, human-robot dialogue data, especially in performing a USAR task, is not readily available and is furthermore rather difficult to collect. A common method widely used for collecting simulated human-computer dialogue data is the Wizard of Oz method. In collecting Wizard of Oz data, a user interacts with a human wizard who pretends to be a dialogue system. Wizard of Oz data is rather not a pure human-computer dialogue and it depends on how freely the wizard can communicate with the user. Alternatively, human-robot USAR dialogue data can be collected by simulating an USAR task in a real setting, such as the experiment in the tunnel accident setting in the previous section. Another option is to model the task with virtual characters in a simulated environment, for instance, in an online multi-player game.

Due to the lack of human-robot dialogue data, we use human-human dialogue data that has a similar setting as human-robot interaction in performing a USAR task (see section 4.1). We use Apollo 17 journal transcripts[1] which describes a collaborative dialogue in exploring an unknown environment by multiple agents [Jones, 1995]. Two of the three dialogue participants in the Apollo 17 dialogue are astronauts working directly on the moon site. The third one, however, monitors the observation on the moon remotely from Earth. This setting is similar to a rescue robot which is sent to a disaster site and operated by a human operator remotely. Since Apollo 17 data reflects partial uncertain observation, it can be used for a basis data. Thus, we try to model a thematic structure of human-robot interactions by using this Apollo 17 data.

A thematic structure for a situated task-oriented human-robot interaction

[1] http://www.hq.nasa.gov/alsj/a17/a17.html

(HRI) should also deal with the challenges in HRI, for instance uncertainty. Uncertainty is ubiquitous and it occurs frequently in HRI. We cannot be certain about the correctness of ASR outputs or about the interpretations of an intention due to system limitations, for example. We discuss about this issue and the differences between human-human and human-robot dialogue data. We show how a thematic structure designed using human-human dialogue data can be used for human-robot interaction scenarios.

1.5 Contributions of the Thesis

The main contributions of this thesis are:

1. introducing a new approach to structuring thematic changes using an probabilistic logic model, namely a Markov Logic Network model, which can be used to describe human-robot interactions,

2. providing a set of logical rules for characterizing the mechanisms of thematic changes and predicting the next theme, especially in a task-oriented dialogue,

3. presenting a way to include intentional perspective in language use to structuring dialogue/discourse themes or the focus of attention.

1.6 Structure of the Thesis

The rest of the thesis is structured as follows.

- Chapter 2 explains the background theories of the work in the thesis. It explains the notion of discourse/dialogue theme and its relation to meaning, belief and intentions. It also describes various approaches to modeling thematic changes and the motivation for using Markov Logic Network in our approach.

- Chapter 3 gives a short review of Markov Networks and First Order Logic. Subsequently, the theory of Markov Logic Network is described.

- Chapter 4 describes the data we used and how we annotated it. It explains how the data is transformed into logical forms serving as inputs for an MLN. The chapter also describes the first-order logic formulas used for generating an MLN model capable of predicting thematic changes. Finally, the inputs and the outputs needed for such an MLN model are illustrated.

- Chapter 5 explains our methodology for evaluating an MLN model. We describe how we split the data into training and test data sets. We show the results of testing the MLN models. We present evaluation results including the performance of an MLN model measure by precision, recall and F1 score in comparison to two baselines. Moreover, we compare the correct predictions of an MLN model to the test annotation, and present an error analysis.

- Chapter 6 discusses how the work in the thesis is able to answer the problems we try to solve, and what the shortcomings of the approach are. We also discuss the application of an MLN model for structuring thematic changes in a human-robot dialogue, especially in performing a USAR task. Finally, we illustrates the applications of an MLN model to help various modules of a dialogue system.

- Chapter 7 presents a conclusion of our work and some discussion about future work.

Chapter 2

Literature

In this chapter, we present a survey of literature related to dialogue themes and building a thematic structure. In section 2.1, we describe the notion of a theme, particularly a dialogue theme. A theme in the dialogue level should not be confused with a theme on the sentence level. Moreover, we describe the notions of meaning, belief and intention in section 2.2. As mentioned in the previous chapter, a theme may express the intention of a dialogue participant, so it can be used for recognizing an intention behind what is said or for predicting the intention of the speaker of the next utterance. In section 2.3, we describe existing approaches that can be used for modeling thematic changes. We also explain the shortcomings of these approaches and the motivation of using a new approach based on Markov Logic Networks.

2.1 Dialogue Theme

Halliday and Matthiessen [2004] described the concepts of theme and rheme in the notion of a clause as a message. Moreover, they explain that a theme is the starting point of the message which gives the clause its context, and a rheme is the rest of the message which elaborates on the theme and contributes to the content of the message. A theme is also referred to as the topic of a sentence and a rheme is the comment about the topic. In the example (2.1.1) below:

(2.1.1) She knits a sweater.

"she" is the theme of the sentence explaining what the sentence is about. Moreover, "knits a sweater" is the rheme of the sentence, that is the comment about the theme. The theme of a clause can be identical to the rheme, as illustrated in example (2.2) below:

(2.1.2) What she knits is a sweater.

where "what she knits" is the theme and "is a sweater" is the rheme. This kind of clause is known as *thematic equative*.

So far in this thesis, the theme and rheme are discussed at the sentence level. At the dialogue level, a theme expresses the main entity or idea of a dialogue at a given point [Hirst, 1981]. The dialogue theme may coincide with a sentence theme, but usually it does not. For instance in (2.1.3), Mary and Jane are talking about flowers:

(2.1.3) Mary: I'm fond of Lavender.
Jane: Lavender smells good.

In the first utterance, the sentence theme is "I" and the discourse theme is "Lavender". In the next utterance, the dialogue theme is unchanged from the first utterance, while the sentence theme turns into "Lavender". Both dialogue and sentences themes are now identical. A dialogue theme represents a topic in a more general scope than a sentence theme. As the dialogue level is higher than the sentence level, a dialogue theme is more general than a sentence theme. Sentential themes and rhemes provide the content or the details of dialogue themes. For example in (2.1.3), the general topic or the dialogue theme is "Lavender". The themes and rhemes of the sentences explain what is said about Lavender, i.e. the first utterance tells us that Mary is fond of it and the second suggests that it smells good.

A dialogue theme is the dialogue participants' focus of attention, which is a small portion of what each of them knows or believes [Grosz, 1981]. A speaker gives clues to the hearer about her current focus and what she wants to focus on next. What is focused on influences what is said, and what is said influences focusing. For instance in (2.1.3), Mary is focusing on Lavender and thus she is saying something about Lavender. Moreover, Mary proposes that the dialogue continues to be about Lavender; either she continues talking about Lavender, or she expects Jane to give a response about Lavender.

Entities being discussed in a dialogue can be represented in a hierarchical associated structure [Sidner, 1979]. An entity in this hierarchy can be selected as the dialogue theme of a dialogue segment (i.e. a unit of a dialogue). Another entity can be selected for another dialogue segment. As a dialogue progresses, the dialogue theme changes within the hierarchy. If (2.1.3) is followed immediately by (2.1.4):

(2.1.4) Mary: What kind of flowers do you like?
Jane: I love Jasmine.

the dialogue theme will change from "Lavender" to "kind of flowers", and then to "Jasmine". In the hierarchical entity structure of the dialogue, the dialogue theme jumps one level up (zoom-out), and then jumps one level down again (zoom-in).

As mentioned in section 1.4, the dialogue themes in a situated dialogue are determined by temporal and spatial aspects. The themes depends on how the dialogue progresses over time, and how the dialogue participants and the environment changes. Moreover, it is also influenced by the dialogue participants' beliefs and intentions about acting in the environment. The spatial setting of a situated dialogue can be represented as a hierarchical graph, which is called a topological abstraction by Zender et al. [2009]. When a dialogue participant moves from one place to another, a new dialogue theme is evoked from the new place. For example, a house contains two bedrooms, a bathroom, a living room and a kitchen. When a robot drives from a living room to a kitchen, it is expected to talk about something in the kitchen, not in the living room. The temporal and spatial aspects seems to account for the structure of a task involving movements in space and time.

The structure of a task-oriented dialogue, which aims at attaining a certain goal, corresponds to the structure of the task or the plan to achieve the task goal. This task structure also gives clues to detect the changes of the dialogue theme. A task typically comprises a number of sub-tasks/plans with sub-goals, sub-sub-tasks/plans with sub-sub goals, and so on. These sub-tasks correspond to sub-dialogues and each sub-dialogue is hold to achieve a sub-goal. In some tasks, the order of sub-tasks can be hard-constrained (i.e. one sub-task must be done before another. For example, to change an inner tube of a bike wheel, one needs to take off the outer tube first, replace the inner tube, and finally put on the outer tube again. The changes in the dialogue theme should be in accordance with the accomplishment of the sub-tasks.

The structure and the execution of a task correspond to joint intentions. A single-agent intention represented by individual utterances can not be expressed purely in terms of joint intention, and vice versa [Levesque et al., 1990]. For example, in a piano duet, each player plays his own part, but they perform the same single musical piece together. Since individual intentions can be different from a joint intention, the task in a task-oriented dialogue is not simply a collection of intentions.

2.2 Meaning, Belief and Intention

Meaning can be explained as a signification (i.e. a significance of something). Grice [1957] distinguished meaning into 2 categories, *natural* and *non-natural senses.* Clark [1996] used different terminologies for these categories, *natural sign* or *symptom* for natural sense, and *signal* for non-natural sense. Although not all uses of meaning can be easily divided into one of these categories, Grice argued that in most cases, our uses of meaning fairly strongly incline to incorporate the use of one of these two categories.

Grice described natural sense as the meaning conveyed naturally by something or the natural significance of something. Natural sense gives evidence of the thing it describes. For example, the high body-temperature of Rachel gives evidence or shows a symptom that she has a fever. On the other hand, non-natural sense is a meaning of something which does not really come naturally. Instead, somebody has an idea of the meaning in mind and expresses it by giving a signal (e.g. an utterance, a gesture) which conveys their intention behind the signal. For example, Keith called Rachel and asked her to go out. Rachel gave a signal to Keith by saying “I have a fever” and by that, she meant that she could not go out and needed a bed-rest.

Grice further divided *non-natural sense* into what Clark called *speaker’s meaning* and *signal meaning.* Speaker’s meaning is what a speaker means or intends with what he is presenting, while signal meaning is what a signal means. The difference between speaker’s meaning and signal meaning is not obvious in English since the same word “mean” expresses 2 distinct senses. However, the difference is straightforward in other languages such as Dutch, where speaker’s meaning is “bedoeling” and signal meaning is “betekenis”, and in German, which has the words “Gemeintes” and “Bedeutung”, respectively. In our previous example, what Rachel meant exemplifies a speaker’s meaning and the signal meaning is what is explicitly meant by her utterance “I have a fever”, that is “she has a fever” or “she is in a condition of having a high body-temperature (i.e. a signal or natural sense/sign)”. A speaker’s meaning is intriguing as it might not be identical to a signal meaning. This relates to the study of pragmatics, where speaker's meaning should be interpreted in the context of the discourse.

A signal is important in a discourse because it is used to achieve a participant's goal and a joint-goal. A speaker introduces a shared basis to be added to the speaker's and the hearer's common ground by means of a signal. The speaker's meaning represents the intention of the speaker by presenting a signal, e.g. saying or doing something. The speaker also intends that her hearer will induce a belief from what she is saying, and she intends that her utterance is recognized as intended. Grice puts this notion in the following way:

> "A meant something by x" is roughly equivalent to "A uttered x with the intention of inducing a belief by means of the recognition of this intention."

If the recognition does not play its part in inducing the belief as it is intended by the speaker, the speaker's intentions may not be fulfilled. For example, Keith may fail to recognize Rachel's intentions and gives a reply such as "What do you mean you have a fever? You were just fine this morning.". Furthermore, the hearer may recognize the speaker intention, but refuse to comply. For example, Keith may give a response to Rachel "You'll get better soon, if you take a fever medicine now. So, we can still go out tonight."

The intention of a speaker reflect her beliefs and the purpose of her utterance. Intentions also describe what is being talked about by a speaker - her focus of attention. The belief recognized by a hearer is important for establishing a common ground between a speaker and a hearer. This common ground accounts for determining a focus of a conversation or a dialogue theme. When a speaker and a hearer have mutual beliefs (e.g. the hearer believes that the speaker believes x), they are likely to talk about the same thing. When they agree on talking about something at a given time, it becomes the shared-focus of attention, the discourse theme, at that point in time.

Illocutionary Acts	Description	Actions
Assertive	The speaker attempts to get the hearer to believe a proposition (something being the case)	comment, suggest, boast, conclude, notify, predict
Directive	The speaker tries to get the hearer to do something in the future	ask, order, request, invite, advise, command
Commissive	The speaker commits to do something in the future	promise, plan, bet, offer
Expressive	The speaker express a feeling or emotion (physiological state)	thank, apologize, congratulate
Declarative	The speaker change or determine a state of the world or the reality	fire, promote, baptize

Table 2.1: Searle's categories of illocutionary acts

A speaker gets her hearer to recognize her intention by presenting a signal, i.e. taking an action. Austin [1975] explained the acts of a speaker in order to get the hearer to recognize the speaker's meaning as *illocutionary acts*, and the speaker's acts to get a hearer to do something as *perlocutionary acts.* The hearer's recognition or understanding of speaker's illocutionary acts is called *illocutionary effect.* The hearer's response to the speaker's perlocutionary acts is called *perlocutionary effect* or *perlocution.* These acts are more generally designated as *speech acts,* i.e. linguistic acts performed by a speaker to express his intentions behind his utterance [Searle, 1969]. Searle [1975] criticized Austin's work and proposed an alternative structure of illocutionary acts. He classifies them into five categories, which are summarized in Table 2.1. Nevertheless, Clark [1996] argued that the scheme has many problems. He argued that the categories do not generate all potential illocutionary acts. Moreover, every illocutionary act only belongs to a single category, whereas an utterance may encode more than one act.

Traum and Hinkelman [1992] introduced their theory of *conversational acts*, which is more general than the speech act theory. In addition to the traditional speech acts, conversational acts include *turn-taking*, *grounding*, and higher level *argumentation acts.* Turn taking is terminology for a dialogue participant taking a dialogue turn. Conversational acts specify acts such as "take turn" and "keep turn" for turn taking. Conversational acts for grounding include "initiate", "continue", "acknowledgement", and "repair". Argumentation acts are higher level acts which combine traditional core speech acts such as "inform", "WH-question" and "Y/N-question". For example, Q&A (Question and Answer) is a commonly used argumentation act for collecting information, and it combines "WH-question" or "Y/N-question" with "answer" acts. Other argumentation acts such as "elaborate", "summarize" and "clarify" include "inform". Conversational acts denote a set of joint speaker-hearer actions. The performance of the actions explains the meaning of a conversation and the meaning is grounded to the satisfaction of both participants.

Modality	Example
Declarative	The weather is nice.
Yes/No question	Did you order the book? Could you pass the salt?
WH question	Why is it the case?
Imperative	Close the window!
Exclamatory	I'm sorry! What a pity!

Table 2.2: Sentence modality in English

Furthermore, a speaker gets the hearer to recognize her speech act by using a certain sentence modality. Sentence mood describes the use of a sentence representing the propositional content and role in a discourse by using linguistic means [Zaefferer, 1990]. Five modalities in English are illustrated in Table 2.2. Declarative mood can be used to assert something (i.e. assertive act) and exclamatory mood to perform an expressive act. A speaker can use Yes/No questions or imperative moods to ask or command (i.e. directive act) the hearer to do an action.

2.3 Modeling Thematic Changes

A thematic structure of a dialogue should represent the relationship between the themes of different dialogue segments. The relationship between a theme in a dialogue segment and a theme in the next immediate dialogue segment is called a thematic change. A theme of a dialogue segment can be identical to or different from another dialogue segment. A thematic structure should demonstrate how a dialogue theme changes from one dialogue segment to another.

We are interested in a thematic structure which is not only able to represent a thematic change, but also to predict the theme of the next discourse segment. As mentioned in section 1.2, predicting the next theme is different from identifying the next theme. When identifying the next theme, we can observe the next utterance and try to identify whether its theme is a new theme or not. If the theme is not new, we try to resolve it to its referent. The task of identifying the next theme is similar to anaphora resolution, when the theme in the next utterance is expressed as a pronoun. When predicting the next theme, on the other hand, we are not informed about the next utterance at all. We only used the information from the utterances in a dialogue history to predict what the theme in the next utterance could be.

An approach to identifying a dialogue theme is described in section 2.3.1. We also describe several approaches to structuring discourse/dialogue theme. Different approaches structure the discourse themes in different ways. In section 2.3.2, we describe a stack model which structures discourse themes based-on the dominance hierarchy of discourse segment purposes. Subsequently, in section 2.3.3, we describe an information status model which explains the status of a theme with respect to a hearer or a discourse. In section 2.3.4, we explain about centering theory which models the salient properties in a discourse. We also discuss Segmented Discourse Representation Structure (SDRS) as a thematic structure in section 2.3.5, and knowledge-based approaches in section 2.3.6. Finally, we discuss the motivations of using Markov Logic Network (MLN) in this work.

2.3.1 Theme Identification

A discourse theme can be identified by using linguistic clues that may be given explicitly by certain words, or derived from sentential structure or rhetorical relationships between sentences [Grosz, 1981]. Moreover, it can be derived from shared knowledge about the relationships among the entities being discussed. Sidner [1979] described an algorithm to determine a discourse theme by using thematic relations and syntactic structure. Some possible discourse themes, or *defaults*, based on semantic categories (i.e. a group of interrelated words defining their meaning) are selected initially and then the expected discourse theme is predicted. Although the prediction of the discourse theme can be wrong, Sidner claimed that once the false prediction is recognized, the true discourse theme can be found easily. She argued that the next sentence can either confirm or reject the expected discourse theme.

Cleft, *pseudocleft* and *there-insertion* sentences explicitly single their discourse themes out. The discourse themes are italicized in the examples below:

1. It was *John* who won the Nobel prize. (cleft-agent)

2. It was *the Nobel prize* that John won. (cleft-object)
3. The one who won the Nobel prize was *John*. (pseudocleft-agent)
4. What John won was *a noble prize*. (pseudocleft-object)
5. There was *a researcher who won a noble prize*. (agent)
6. There was *a noble prize which John won*. (object)

A pronoun is a good indicator referring to the focus of the previous sentence. Syntactic positions, such as subject or object positions, also provides a clue to identifying a discourse theme. The object of an action is the default position among other verb positions for the expected discourse theme. In the example (2.3.1) below, "a cake" is the direct object and the expected discourse theme. The pronoun "it" in the second sentence co-specifies "a cake" in the first sentence.

(2.3.1) Jenn baked a cake. She gave it to Nick.

Another clue is a thematic relation, which is a relation between a noun phrase and a verb, and a thematic position, i.e. the verb/noun phrase position representing the relationship of being affected by the action of the verb. A thematic position appears most commonly as a direct object, but it may also appear in other positions including an instrument, a goal, a location, and so on. For example in (2.3.2),

(2.3.2) The car crashed into a tree.

"the car" is in the thematic position (i.e. as both subject and agent) and it is the expected discourse theme. Agent is the last preferred thematic position since it is not typical to be the main topic of a discussion, unless no other entities have been mentioned.

The following is the expected discourse theme algorithm described in Sidner [1979]. An expected discourse theme is chosen as follows:

1. The subject of a sentence for an *is-a* or *there-insertion* sentence.
2. The first element of a Default Expected Focus (DEF) list has this order: thematic relation, other thematic positions with the agent last, the verb phrase.

2.3.2 Stack Model

Grosz and Sidner [1986] discuss a dialogue theme as being the dialogue participants' focus of attention. Moreover, they describe a thematic structure as an attentional state - the abstraction of the focus of attention. An attentional state is modeled as a stack of focus spaces containing the representations of entities in the focus. A focus space also includes a discourse segment purpose (DSP) indicating the intention conceived in a segment of a discourse. The dialogue themes of dialogue segments are structured in a focus stack, where its entries are ordered by recency. The most recent theme is put on the top of the stack and the oldest theme being talked about is found at the bottom of the stack.

Two kinds of relations between DSPs, namely *satisfaction-precedence* and *dominance* relations, are used for structuring the focus spaces in a stack model. *Satisfaction-precedence* relation between two DSPs shows the order of DSP satisfaction. Before satisfying a DSP X, another DSP Y may be required to be satisfied first. In other words, the satisfaction of the DSP X requires the satisfaction of DSP Y. The DSP X is hence said to *dominate* the DSP Y (i.e. *dominance* relation). Focus spaces are pushed into and popped out of a focus stack with respect to these two structural relations. A focus space is pushed into the a focus stack when a DSP of a discourse segment contributes to the DSP of the immediately preceding discourse segment. On the other hand, a focus space is popped from the stack when a DSP of a discourse segment contributes to some DSP higher in the dominance hierarchy of the DSPs. Before a new focus space can be inserted into a focus stack, several focus spaces have to be popped.

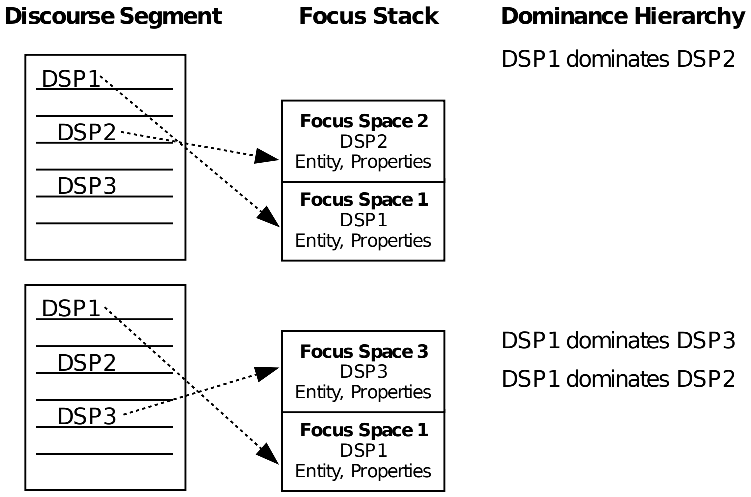


Figure 2.3.1: An illustration of the mechanism of a stack model

A stack model is appropriate to model a very well-structured task-oriented dialogue where the tasks have to be carried out in a certain order. If a dialogue progresses according to this order, the DSPs of the dialogue segments can be represented in a dominance hierarchy nicely. In practice, however, a dialogue may contain dynamic thematic/focus changes, where the foci in the dialogue and the DSPs of the dialogue segments are intertwined with each other. A stack model is not flexible for modeling such cases because the stack model cannot contain more than one focus space per level in a dominance hierarchy. To talk about another focus at the same dominance level, the previous focus at that level has to be popped first. This mechanism is illustrated in Figure 2.3.1. Focus space 2 and focus space 3 share the same level. Once the dialogue participants stopped talking about focus space 2, it is popped out of the stack and then focus space 3 can be pushed to the stack. Moreover, in the stack mechanism, a focus space which has been popped from a stack is no longer in the structure. Therefore, its re-occurrence in a dialogue cannot be explained by the model.

2.3.3 Information Status Model

Information status models the status of a theme being old or new. The notion of information status can be categorized into hearer-old/new and discourse-old/new regarding where the information status is considered [Prince, 1992]. A theme is considered as hearer-new, if it has not been known to a hearer yet. On the other hand, it is considered as a discourse-new, if it has never been introduced in a discourse so far.

Information status in the hearer's head depends on the speaker's beliefs about the hearer's beliefs. For instance, in the short dialogue example (2.1.3) Mary (the speaker) believes that Jane (the hearer) knows Lavender (i.e. Lavender is not something new to Jane). Therefore, Lavender is considered as hearer-old (i.e Mary assumes Lavender is old information to Jane). On the other hand, "I'm fond of" can be considered as hearer-new when the speaker assumes that it has not already been known to the hearer. Hearer old/new accounts for grounding, which is essential for establishing a discourse theme. As described in the section 1.1, a collaborative dialogue requires mutual beliefs and understanding of what is being talked about.

Information status can also be seen from the point of view of a dialogue history being constructed during dialogue processing. In this case, Lavender is discourse-new because it is a new piece of information in the dialogue history. As the dialogue progresses to the second utterance, Lavender turns out to be discourse-old since it has been mentioned before. Discourse old/new gives us a clue about the status of a theme in a discourse.

Information status explains the new/old status of a theme. It explains how the status of a theme changes from hearer-new to hearer-old, as well as from discourse-new to discourse old. However, it does not explain how to predict such status changes. It does not show when a theme is likely to be hearer-old/new or a discourse-old/new theme in the next utterance.

2.3.4 Centering Model

Centering theory provides a framework to model the salience properties in a discourse. It adopts Sidner [1979]'s focusing algorithm (see section 2.3.1) to conduct the centering process. In centering theory, the entities which connect one utterance to the other utterances are designated as *centers*. The centers in an utterance are the elements of a set of *forward looking centers* of that utterance. The *forward looking centers* are partially ordered to reflect the relative salience in the corresponding utterance. The centers are ranked in the order in which they are listed. Moreover, the centers are ranked according to their grammatical position [Grosz et al., 1995]. Subject position is ranked higher than object position, which is ranked higher than other position. A subject is considered to be more likely to be salient and to appear in the next utterance. Besides grammatical position, information status can also be used to rank the centers. Strube [1998] suggests that a hearer-old center should be ranked higher than a hearer-new center.

A single *backward-looking center* in the next utterance U_{n+1} connects with one of the forward looking centers. The *backward-looking center* is the most highly ranked center in an utterance that is realized in the next utterance. The more highly ranked a center in a set of *forward looking centers* is, the more likely it is to be a *backward-looking center.*

Centering theory models the relationship between the centers of two consecutive utterances. It specifies the relationship between the *backward-looking center* of the next utterance $C_b(U_{n+1})$ and the *backward-looking center* of the last utterance $C_b(U_n)$. Moreover, it relates the *backward-looking center* of the next utterance and the most highly ranked center in the next utterance $C_p(U_{n+1})$. Three types of center transitions between two consecutive utterances are defined as follows:

> CENTER CONTINUATION: $C_b(U_{n+1}) = C_b(U_n)$ and $C_p(U_{n+1}) = C_b(U_{n+1})$. A center is continued when it is realized in the next utterance and is likely to be realized in subsequent utterances (U_{n+2}).
>
> CENTER RETAINING: $C_b(U_{n+1}) = C_b(U_n)$, but $C_p(U_{n+1}) \neq C_b(U_{n+1})$. A center is retained when it is realized in the next utterance, but it is likely to be changed in the subsequent utterances.
>
> CENTER SHIFTING: $C_b(U_{n+1}) \neq C_b(U_n)$. A center is shifted when it is neither continued nor retained.

Using these center transitions, centering theory can structure thematic changes in a discourse. A dialogue theme can be considered as the most salient center in an utterance. Strube and Hahn [1999] specifies that $C_p(U_n)$, the highest ranked center in a set of *forward looking centers* of an utterance, is what the utterance is about. It is the focus of attention, or the dialogue theme, in that utterance. Centering theory can predict the theme in the next utterance by using the *backward-looking center.* That means it always continues the theme of the last utterance.

Strube [1998] introduced an algorithm combining centering theory and information-status. Strube's approach is based on functional information structure, so it is called functional centering. In functional centering, the set of possible centers is ranked according to their information status. Particularly, hearer-old centers are ranked higher than hearer-new themes. The set of possible centers contains the salient discourse entities in the next and the last utterance. Strube illustrated his approach for anaphora resolution. Every time an anaphoric expression is encountered, anaphora resolution using functional centering is solved by testing the elements of the possible center set, until the resolution succeeds. This approach is useful for identifying a theme in the next utterance, but it is not able to predict the theme.

2.3.5 Segmented Discourse Representation Structure

Lascarides and Asher [2007] described an approach to structuring a discourse using what they entitled Segmented Discourse Representation Theory (SDRT) and dynamic semantics. SDRT uses a logical representation of syntax and semantics of the language and rhetorical relations between utterances/discourse segments to model the semantics or pragmatics interface. SDRT provides a logic for representing and interpreting the logical forms of a discourse (i.e. logic of information content), and a logic for constructing logical forms (i.e. glue logic). The logic of information content defines a Segmented Discourse

Representation Structure (SDRS), which is the discourse structure. It relates speech act discourse referents (i.e. labels of discourse segments) by using discourse relations (e.g. elaboration, contrast, result).

An SDRS implements dynamic semantics to represent the meaning of a discourse segment as a relation between an input and an output discourse context. Dynamic semantics observes the effects of logical structure on various kinds of anaphora (i.e. the use of one word to substitute another preceding word), such as pronoun, tense and presupposition, in pragmatic phenomena. An SDRS enables anaphora resolution. It can be used for recognizing, but not predicting the theme of the next discourse segment.

By using discourse relations, SDRS can represent the relationship of the themes of the discourse segments. Furthermore, we can model the thematic changes throughout a discourse. However, an SDRS works for systems which require highly structured discourse context, and therefore, it is deemed to be too complicated for other systems which only use simple structures. Besides, an SDRS does not seem to be robust enough for noisy data. For instance, the relationships among discourse segments in an SDRS cannot be well-represented for noisy data. Thus, SDRS cannot model thematic changes in noisy discourse data.

2.3.6 Knowledge-Based Model

To structure a dialogue, a knowledge-based model makes use of an ontology of entities, activities or plans. In performing an activity or a task, people often use common knowledge, and their actions typically follow a certain pattern. Such common knowledge and typical patterns can be described in a knowledge base. For example, the appointment scheduling described in a section 1.1 reflects a pattern how people usually make appointments. People usually agree on a date first and then, on a time on that date.

The speech-to-speech translation system VERBMOBIL [Wahlster, 2000] accommodates cross-lingual appointment scheduling. COSMA (Cooperative Schedule Management Agent) was also designed to assist in scheduling appointments among multiple participants [Busemann et al., 1994]. Both VERBMOBIL and COSMA use a knowledge-based approach to model how a dialogue might progress in scheduling an appointment. VERBMOBIL makes use of a taxonomy of a few speech acts (e.g. suggest, accept and reject) to track a negation process in making an agreement on a place and date. Moreover, it uses an ontology of activities and plans. Plans in this ontology describe objects (e.g. locations, temporal expressions), situations (e.g. events and activities) and qualities (i.e. features of an object or situation) in the appointment scheduling domain. The concepts in the ontology are related to each other. For instance, the concept *traveling* is related to the concepts *transportation* and *lodging*. Knowledge-bases are used in different modules to predict the next dialogue act or a potential sequence of dialogue acts. They are also used to construct an intentional structure describing the plan for scheduling an appointment.

A knowledge-based model provides information on how a dialogue might develop. Therefore, it can be used to predict thematic changes in a dialogue. Furthermore, it gives us the advantage of explaining why a dialogue develops in a certain direction. However, the approach suffers from some limitations as mentioned in section 1.2. Firstly, a knowledge-based approach is rather domain specific and thus, different ontologies are needed to structure dialogues in different domains. Secondly, the scalability of a knowledge-base is limited by its complexity. A knowledge-based model can deal with simple interactions, but tends to fail at coping with more complex interactions. For instance, the COSMA and VERBMOBIL systems work well to merely arrange a place and time. However, they may fail quickly when people start talking about the best way to reach the meeting location.

2.3.7 Discussion

In the previous sections, we described different approaches to building a thematic structure. These approaches model a dialogue theme and its changes in different ways. Each approach has its advantages and disadvantages. A stack model is able to model how a theme changes to another theme throughout a discourse. Moreover, it models the theme and the thematic changes based on the purpose of a discourse segment or the intention behind it. This approach works for very well structured and ordered task-oriented dialogues. However, it is considered inflexible because the theme prediction is restricted by the dominance hierarchy of the themes. On the other hand, the information status model is able to represent the status of a theme as being new or old information to a hearer or in a discourse. However, it does not explain how to predict the status changes of a theme.

Centering theory explains possible thematic changes of a theme. It can be used for both recognizing and predicting a thematic change, but it always predicts a continuation of a theme from the last utterance. In centering theory, the prediction only depends on the last utterance and a centering model simply continues the most highly ranked theme in the last utterance at all times. It cannot predict multiple themes which can actually be expressed in a single utterance. Since centering theory only consider the theme of the last utterance, it cannot model a re-occurrence of a theme from an utterance prior to the last utterance.

An SDRS is a rich discourse structure which can model the relationship between the themes of two utterances. It also enables anaphora resolution which can be used for recognizing the theme of the next utterance. However, it cannot really predict an appearance of a new theme in the next utterance. Moreover, it does not seem to be robust enough to model the thematic changes in noisy dialogue data.

A knowledge-based approach seems to be promising, but it is domain specific and cannot deal with complex dialogue interactions. To deal with the drawbacks of the existing approaches, we introduce a new approach using a Markov Logic Network (MLN) model which uses both a knowledge base and statistics. An MLN model combines logic and statistics to determine the tendency of a logical rule and the probability of a prediction. By means of a set of weighted logical rules, an MLN model can predict a continuation of a theme from the last utterance or the occurrence of a new theme. Because it does not only consider the last utterance, but also all other utterances in a given dialogue, it can also predict re-occurrences of themes older than the theme of the last utterance.

An MLN model is more flexible than a stack model because it always considers all the themes of the previous utterances. It can cover more possibilities of dialogue moves than a knowledge-based model mainly guided by an ontology. Since an MLN is not restricted by a certain pattern of dialogue progression, it can deal with dynamic dialogue progression in which the thematic changes do not follow a certain pattern. A knowledge-based model, on the other hand, is constrained by the ontology it uses. It tends to fail at explaining dialogues that are not well-structured or progress in a certain direction, but often jump from one concept to a not closely related concept. Moreover, an MLN seems to be robust to noisy data because it computes appearance tendencies of its logical rules throughout a given dialogue data and models thematic changes based on weighted logical rules.

Chapter 3

Markov Logic Network

In this chapter, we introduce a new approach to structuring a dynamic dialogue using a Markov Logic Network (MLN) model. MLNs take advantages of both logic and statistics. MLNs make use of uncertain knowledge expressed in first order logic rules to make inference, and the uncertainty is handled by a probabilistic approach. The mechanisms of thematic changes can be transparently specified in an MLN as first-order logic. Thus, MLN models can be used to predict the dialogue themes/foci in the next utterance.

The first-order logic rules in an MLN create a network of probabilistic dependency, which is a Markov network. Markov networks provide a way to represent the joint distributions of its nodes. In an MLN, the nodes are each possible grounding of formulas in first-order logic rules. An explanation of Markov networks is given in Section 3.1. Section 3.2 describes the theory of First-Order Logic, and finally, we explain MLNs in section 3.3.

3.1 Markov Networks

A Markov network (also known as a Markov random field) is a dependency model of context-dependent entities [Pearl, 1988]. It follows a Markov property (i.e. memoryless property) which specifies that the future states only depend on the present state, and that the past states are irrelevant. The present state in a Markov network is defined by means of random variables representing an interconnected network of context-dependent entities.

A Markov network is represented as an undirected graph whose edges designate probabilistic dependencies of every adjacent node. Each node in the graph represents a random variable and each clique (i.e. a subset of the graph where every two nodes are adjacent to one another or connected by an edge) is associated with a potential function characterizing the state of the clique (i.e. clique potential).

The representation of dependencies in a Markov network is similar to that in a Bayesian network [Pearl, 1988]. Markov and Bayesian networks are different from each other in terms of the dependencies they can represent. Markov networks can represent circular dependencies (i.e. no valid order of which node should come first), while Bayesian networks cannot. On the other hand, Bayesian networks can represent induced and non-transitive dependencies which cannot be represented by Markov networks. Since a Markov network connects two independent variables directly, it does not represent any independence.

The dependencies among nodes in a Markov network is characterized by the joint distribution of the set of the random variables $X = (X_1, X_2, \ldots, X_n)$. The probability of the joint distribution represented by a Markov network is given by [Richardson and Domingos, 2006]

$$P(X = x) = \frac{1}{Z} \prod_k \phi_k(x_k) \tag{3.1.1}$$

where ϕ_k is a potential function, that is a non negative real valued function of the state of the k-th clique. Moreover, x_k is the state of variables in the k-th clique and $Z = \sum\limits_{x \in \chi} \prod\limits_k \phi_k(x_k)$ is a normalizing constant called the *partition function*. χ is the set of all possible states of variables in the Markov network.

A Markov network can be represented as a log linear model by computing the clique potential as an exponentiated sum of weighted features of the state as follows.

$$P(X = x) = \frac{1}{Z} exp \left(\sum_k w_k f_k(x_k) \right) \tag{3.1.2}$$

where the weight w_k of a possible state x_k is $log \phi_k(x_k)$ and $f_k(x_k) \in \{0, 1\}$ is the binary feature for x_k[1]. Since the formula will grow exponentially with respect to the size of the cliques, the number of features should be minimized, for instance by specifying a logical function of the state of a clique.

The Markov blanket of a node is the minimal set of nodes which enables it to be independent of the remaining network. A Markov blanket of a node in a Markov network is the set of its neighboring nodes.

3.2 First Order Logic

First-order logic or first-order predicate logic is a formal language which can be used to represent objects in the real world and the relations between those objects. By using first order logic, we can make inferences, which is to say, we can derive logical conclusions from known premises (propositions) or premises that are assumed to be true. First-order logic is similar to propositional logic, but unlike propositional logic, quantifiers are used in first-order logic. First-order logic is used as the standard formal logic for describing axioms.

The syntax of first-order logic uses symbols of three types, namely a function, a predicate and a variable [Gamut, 1991]. A function maps a tuple of objects to an object and a term is a function that has zero arguments. A term can be a constant or a variable in a language. Constants refer to objects in the domain of interest and variables are symbols which range over these objects. We write variables in lowercase and the first letter of constants in uppercase. A formula in first-order logic can be constructed by using terms and predicates. A predicate represents a relation among objects (e.g. *own* signifies a binary relation of some object possessing some other object) or a property of an object (e.g. *pet* signifies the property of an object being a pet).

[1] A feature may be specified differently, e.g. a real-valued function of the state.

Text	First-Order Logic	Clausal Form
A pet is owned.	$\forall x\, pet(x) \rightarrow \exists y\, own(y, x)$	$\neg pet(x) \vee own(y, x)$
Somebody who owns a pet, feeds it.	$\forall x \forall y\, pet(x) \wedge own(y, x) \rightarrow feed(y, x)$	$\neg pet(x) \vee \neg own(y, x) \vee feed(y, x)$

Table 3.1: First-Order-Logic formulas

Interpretations of Terms
$[\![\alpha]\!]^{M,g} = V_M\,(\alpha)$
$[\![\alpha]\!]^{M,g} = g(\alpha)$

Interpretations of Formulas
$[\![P(t_1, ..., t_n)]\!]^{M,g} = T$ iff $\left\langle [\![t_1]\!]^{M,g}, ..., [\![t_n]\!]^{M,g} \right\rangle \in V_M(P)$
$[\![t_1 = t_2]\!]^{M,g} = T$ iff $[\![t_1]\!]^{M,g} = [\![t_2]\!]^{M,g}$
$[\![\neg\varphi]\!]^{M,g} = T$ iff $[\![\neg\varphi]\!]^{M,g} = F$
$[\![\varphi \wedge \psi]\!]^{M,g} = T$ iff $[\![\varphi]\!]^{M,g} = T$ and $[\![\psi]\!]^{M,g} = T$
$[\![\varphi \vee \psi]\!]^{M,g} = T$ iff $[\![\varphi]\!]^{M,g} = T$ or $[\![\psi]\!]^{M,g} = T$
$[\![\varphi \rightarrow \psi]\!]^{M,g} = T$ iff $[\![\varphi]\!]^{M,g} = F$ or $[\![\psi]\!]^{M,g} = T$
$[\![\varphi \leftrightarrow \psi]\!]^{M,g} = T$ iff $[\![\varphi]\!]^{M,g} = [\![\psi]\!]^{M,g}$
$[\![\exists x\varphi]\!]^{M,g} = T$ iff there is at least an $a \in U_M$ such that $[\![\varphi]\!]^{M,g[x/a]} = T$
$[\![\forall x\varphi]\!]^{M,g} = T$ iff for all $a \in U_M$, $[\![\varphi]\!]^{M,g[x/a]} = T$

Figure 3.2.1: Interpretations of Terms and Formulas with respect to a model structure M and a variable assignment g

Definition 1. A term t is a function with zero argument, a constant or a variable.

An *atomic formula* or an *atom* is the simplest formula. It is composed by a predicate and its arguments, where each of the arguments is a term (e.g. $own(y, x)$ and $pet(x)$). A *positive literal* is an atomic formula and a *negative literal* is a negation of an atomic formula. Formulas or well-formed formulas can be constructed recursively from atomic formulas by using quantification over variables, negations and logical connectives (i.e. conjunctions, disjunctions, implications, equivalence).

Definition 2. $P(t_1, ..., t_n)$ is a well-formed formula where $P \in predicate$. If φ and ψ are well-formed formulas, then $\neg\varphi$,$(\varphi \wedge \psi)$,$(\varphi \vee \psi)$,$(\varphi \rightarrow \psi)$,$(\varphi \leftrightarrow \psi)$ are well-formed formulas. If $x \in variable$ and φ is a well-formed formula, then $\forall x\varphi$ and $\exists x\psi$ are well-formed formulas. All atomic formulas are well-formed formulas.

A negation of a formula is a formula (e.g. $\neg pet(x)$). A disjunction of formulas is a formula (e.g $\forall x \exists y\, pet(x) \wedge own(y, x)$). A conjunction of formulas is a formula (e.g. $\forall x\, pet(x) \vee \neg pet(x)$). An implication of formulas is a formula (e.g. $\forall x\, pet(x) \rightarrow \exists y\, own(y, x)$). An equivalence of formulas is a formula (e.g. $own(y, x) \leftrightarrow belongTo(x, y)$). Every first-order logic formula can be converted into a clausal form or conjunctive normal form (CNF). Table 3.1 exemplifies first-order logic formulas and their corresponding clausal forms.

A language L consists of a set of variables $V = \{V_1, ..., V_n\}$, a set of constants $C = \{Anna, Dog\}$, and a set of predicates $P = \{pet/1, own/2\}$.

The Herbrand Universe and Herbrand base of the language L are
$HU = \{Anna, Dog\}$
$HB = \{pet(Anna), pet(Dog), own(Anna, Dog), own(Dog, Anna)\}$

A possible Herbrand interpretation of the language L is
$D = \{Anna, Dog\}$
$F = \{Anna \rightarrow Anna, Dog \rightarrow Dog\}$
$R = \{pet(Dog), own(Anna, Dog)\}$

Figure 3.2.2: A possible Herbrand interpretation

A term or a formula does not have a semantic meaning unless it is given an *interpretation.* An interpretation is given with respect to a model structure M and a variable assignment g, which assigns a variable to an element of U_M. Figure 3.2.1 depicts the formal definitions of the interpretation of terms and formulas. The variable assignment $g[x/a]$ assigns a to x and assigns the same values as g to all other variables. $g[x/a](y) = a$ if $x = y$ and $g[x/a](y) = g(y)$ if $x \neq y$.

Definition 3. $M =< U_M, V_M >$is a model structure when U_M is a non-empty set and V_M is an interpretation function. $V_M(P) \subseteq U_{M^n}$ where P is a n-ary predicate and $V_M(c) \subseteq U_M$ where c is a constant.

An interpretation of a term assigns the object that is represented by the term. An interpretation of a formula assigns a truth value (i.e. *true* or *false*) to the formula. A formula is satisfiable if it is true under at least one interpretation. $\forall x \varphi$ is true iff φ is true for *every object* x in the domain. $\exists x \varphi$ is true iff φ is true for *at least one object* x in the domain. A negation of a formula is true iff (i.e. if and only if) the formula is false. A conjunction of formulas is true iff every formula is true. A disjunction of formulas is true iff there is a formula whose value is true. An implication of formulas is true iff the formula in the left hand side is false, or the formula in the right hand side is true. An equivalence of formulas is true iff the formulas have the same truth values.

A *ground term* is a term that does not contain any variable. A *ground atom* or *ground predicate* is an atomic formula where all of arguments are ground terms. A *ground formula* is a formula where all of its arguments are ground atoms. An atomic formula is grounded by replacing its variables with constants.

A Herbrand interpretation is an interpretation which maps a constant to itself and it determines the truth value of a ground atom. Formally, a Herbrand interpretation HI of a language L has a domain D which is the Herbrand universe HU of L (i.e. a set of constants), an identity function F, and a subset R of the Herbrand base HB (i.e. the set of ground atoms) that is true. The number of Herbrand interpretations is the size of the power set of the Herbrand base. An example of a possible Herbrand interpretation is given in Figure 3.2.2.

3.3 Markov Logic Network

A Markov logic network (MLN) is a probabilistic logic model which combines Markov networks and first-order-logic [Richardson and Domingos, 2006]. It is a combination of knowledge-based and statistical approaches. It uses a knowledge base of logical rules and atomic formulas. Moreover, it consists of a set of weighted first-order logic formulas and the weights are learned from a relational database of ground atoms. In the following section, we describe the definition of an MLN. In section 3.3.2, we describe how the weights of first-order logic formulas in an MLN can be learned. In section 3.3.3, we explain how MLN make an inference to answer a query.

3.3.1 Definitions

An MLN consists of a set of weighted first-order logic formulas. The weight attached to a first-order logic formula reflects how strong a formula is as a constraint. First-order logic formulas have the restriction that a formula should be consistent with a Herbrand interpretation. If it is inconsistent, then it has zero probability. MLNs soften this restriction by allowing a formula to be inconsistent with a Herbrand interpretation. The weight of a formula in an MLN represents the difference in log probability between a Herbrand interpretation that satisfies a formula and one that does not. The higher the weight, the greater the difference is.

In an MLN, constants and variables are typed [Richardson and Domingos, 2006]. Typed constants can only represent objects in the specified type. Typed variables can range only over objects of the specified type. For example, Dog, Cat, and $Fish$ are constants in an $Animal$ domain. If the constant Dog is typed as an Animal, it can only represent an animal object. If the variable x is typed as an Animal variable, then it can only range over the $Animal$ domain.

A Markov logic network L is defined formally as a set of pairs (F_i, w_i) where F_i is a first-order logic formula and w_i is a weight. A finite set of constants $C = \{c_1, c_2, ...c_n\}$ represent objects in the domain. Together with C, an MLN L generates a ground Markov Network $M_{L,C}$ with respect to the grounding and interpretation of its predicates. All possible groundings of a predicate in L are obtained by replacing each variable of the predicate once with each constant in C. Different sets of constants will generate different ground Markov Networks. However, the grounding of the same formula in an MLN will have identical weight in its different ground Markov Networks.

Richardson and Domingos [2006] define $M_{L,C}$ as follows.

1. $M_{L,C}$ contains one node for each possible grounding of each predicate in L. If the ground atom is true, then the value of the node is 1. If it is false, then the value is 0.

2. $M_{L,C}$ contains one feature for each possible grounding of each formula F_i in L. The value of a feature is 1, if the ground formula is true, and 0 otherwise. The weight of the feature is the w_i associated with F_i in L.

The ground Markov Network of the set of first-order-logic formulas L in Table 3.1 and the set of constants $C =$ "$Anna$","Dog" is given in Figure 3.3.1. Each node in the ground Markov Network is a ground atom (e.g. $pet(Dog)$).

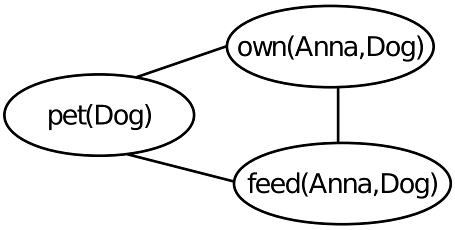


Figure 3.3.1: Ground MLN Example of First-Order Logic Formulas in Table 3.1 and the constants “Anna” and “Dog”

If two atoms appear together in some grounding of a formula in an MLN, then, the atoms are connected with an arc. The arcs can be used to infer a conditional probability of atoms (e.g. the probability of Anna feeds Dog given Dog is a pet and Anna owns the Dog).

Each state in $M_{L,C}$ (i.e. a state from given by the nodes and their truth values) represents a possible Herbrand interpretation. The probability distribution over a possible Herbrand interpretation x specified by the ground Markov network $M_{L,C}$ is given by

$$P(X = x) = \frac{1}{Z} exp(\sum_i w_i n_i(x)) \tag{3.3.1}$$

where $n_i(x)$ is the number of true groundings of F_i in x. The ground atoms in MLNs are thus either 0 or 1, which makes it difficult to model observation uncertainty; see also [Lison et al., 2010].

3.3.2 Weight Learning

Weights for the set of first-order logic formulas in MLNs are learned from databases of ground atoms. If a ground atom is not in the databases, it is assumed to be *false*. A database is a vector of possible ground atoms $x = (x_1, x_2, ..., x_n)$. x_k is the truth value of the k-th ground atom where $1 < k < n$. The value of x_k is 0, if the k-th ground atom is a negative literal in the database or if it does not appear in the database at all. Otherwise, the value is 1.

Given a database x, MLN weights can be learned by computing the derivative of the log-likelihood in equation 3.1.2 with respect to its weight. The derivation is given by [Richardson and Domingos, 2006]

$$\frac{\partial}{\partial w_i} \log P_w(X = x) = n_i(x) - \sum_{x'} P_w(X = x') n_i(x') \tag{3.3.2}$$

where $n_i(x)$ is the number of true groundings in the database x for the i-th formula. χ is the set of all possible databases. $P_w(X)$ is the probability of the joint distribution of ground atoms using the weight vector $w = (w_1, ..., w_i, ...)$. The i-th component of the gradient is the difference between the number of true groundings of the i-th formula in the database and its expectation.

Richardson and Domingos [2006] argued that Equation 3.3.2 is not efficient due to an intractable counting of the number of true groundings of a formula in a database and the expected number of true groundings. A more efficient approach is optimizing the pseudo-likelihood

$$P^*_w(X = x) = \prod_{k=1}^{n} P_w(X_k = x_k|M\,B_x(X_k)) \tag{3.3.3}$$

where $M\,B_x(X_k)$ is the state of a Markov blanket (i.e. the truth values of the neighboring nodes) of X_k in the database. The gradient of the pseudo-log-likelihood is

$$\begin{aligned}\frac{\partial}{\partial w_i}\log P^*_w(X = x) = \prod_{k=1}^{n}&[n_i(x) - P_w(X_k = 0|M\,B_x(X_k))n_i(x_{X_k=0})\\ &-P_w(X_k = 1|M\,B_x(X_k))n_i(x_{X_k=1})]\end{aligned} \tag{3.3.4}$$

where $n_i(x_{X_k=0})$ is the number of true groundings of the i-th formula, when X_k is forced to be 0, and similarly for $n_i(x_{X_k=1})$.

3.3.3 Inference

Formulas in an MLN are typically converted into a more regular form, such as clausal form or conjunctive normal form (CNF), to carry out an automatic inference. An MLN can answer queries which ask for the probabilities of formulas to hold, given that other formulas specified in an MLN hold. In other words, the queries ask how probable it is that the set of first-order logic formulas in an MLN entails the queries.

The probability of formula F_1 holds, given that formula F_2 holds in $M_{L,C}$, is [Richardson and Domingos, 2006]

$$\begin{aligned} P(F_1|F_2, L, C) &= P(F_1|F_2, M_{L,C}) \\ &= \frac{P(F_1 \wedge F_2|M_{L,C})}{P(F_2|M_{L,C})} \\ &= \frac{\sum_{x\epsilon\chi_{F_1}\cap\chi_{F_2}} P(X = x|M_{L,C})}{\sum_{x\epsilon\chi_{F_2}} P(X = x|M_{L,C})} \end{aligned} \tag{3.3.5}$$

where $\chi_{F_1}$is the set of Herbrand interpretations where F_1 holds and $P(x|M_{L,C})$ is given by equation 3.3.1. $P(F_1|F_2, L, C)$ can be estimated by using an Markov Chain Monte Carlo (MCMC) algorithm. which does not allow any transition to the states where F_2 does not hold, and calculates the number of samples in which F_1 holds.

An MCMC algorithm generates samples using a Markov chain mechanism. In a Markov chain, the transition probabilities between sample values are only accounted by the most recent sample value. Examples of MCMC methods are Gibbs sampling and slice sampling. Since the computation of an MCMC algorithm is likely to be too slow for arbitrary formulas, Richardson and Domingos [2006] proposed an inference algorithm for the case where F_1 and F_2 are conjunctions of ground atoms. They argued that ground atoms are most commonly queried in practice and that the algorithm for such queries is more efficient than a direct application of Equation 3.3.5.

Chapter 4

MLN-based Thematic Structure

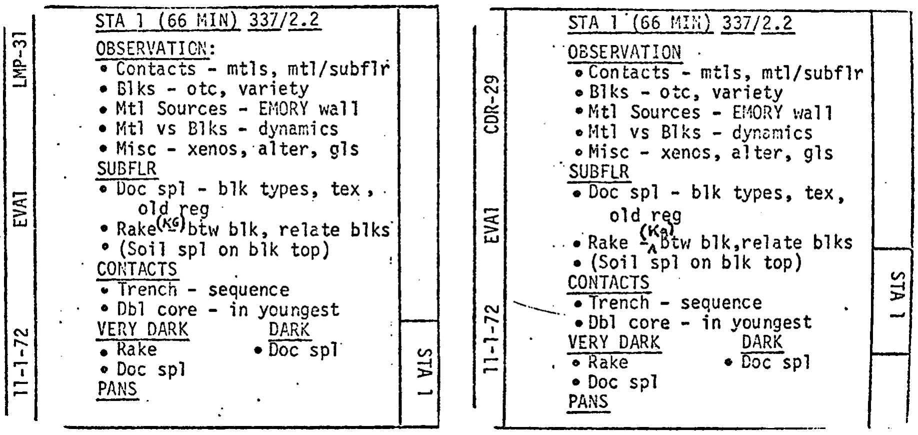
We attempt to build a thematic structure that is able to describe the thematic changes in a human-robot conversation, especially in performing a USAR task. However, as mentioned in 5.4, due to the lack of human-robot dialogue data, we make use of the Apollo 17 transcripts which contain human-human dialogue data. In section 4.1, we describe the similarity between the data we use and a typical human-robot USAR dialogue. We also explain how we annotate the data. Then, we present a set of first-order logic rules to construct an MLN model, in section 4.2.1. The hand-crafted logical rules describe the mechanisms behind thematic changes observed from the Apollo 17 transcripts. These rules enable an MLN-based thematic structure to predict the next dialogue theme.

4.1 Data

To structure the dialogue theme of a human-robot conversation, we should ideally observe a human-robot dialogue. However, human-robot dialogue, especially in performing a USAR task, is not readily and easily available. Moreover, it is difficult, time consuming, and expensive to collect, because it requires human experts in the USAR area, a physical robot and a robotic dialogue system that is able to properly communicate with a human. In practice, human-robot dialogue data is typically needed in a development stage, while the robot is still being developed. Alternatively, human-robot dialogue data can be collected by using the wizard of Oz technique or simulating the real task. However, the collected data is not purely human-computer dialogue.

Due to the lack of human-robot dialogue data, we use human-human dialogue data which is usable for the purpose of this work. The data collected in the tunnel accident simulation described in section 1.3 is a conversation between human rescuers. Although it does not demonstrate human-robot interactions, it explains human behavior in performing a USAR task. This kind of data could be used as a starter to model thematic changes in human-robot interactions. Unfortunately, it only contains a short conversation (approximately 7 minutes) and therefore it is not sufficient to be used in this work.

Alternatively, we use the Apollo 17 journal transcripts which describe the Apollo 17's mission to collect sample materials and to deploy some scientific instruments on the moon. The dialogue in the Apollo 17 transcripts involves three people performing a joint activity of observing and taking samples at different



LMP-31 EVA1 11-1-72

STA 1 (66 MIN) 337/2.2

OBSERVATION:
- Contacts - mtls, mtl/subflr
- Blks - otc, variety
- Mtl Sources - EMORY wall
- Mtl vs Blks - dynamics
- Misc - xenos, alter, gls

SUBFLR
- Doc spl - blk types, tex, old reg
- Rake (KG) btw blk, relate blks
- (Soil spl on blk top)

CONTACTS
- Trench - sequence
- Dbl core - in youngest

VERY DARK
- Rake
- Doc spl

DARK
- Doc spl

PANS

STA 1

CDR-29 EVA1 11-1-72

STA 1 (66 MIN) 337/2.2

OBSERVATION
- Contacts - mtls, mtl/subflr
- Blks - otc, variety
- Mtl Sources - EMORY wall
- Mtl vs Blks - dynamics
- Misc - xenos, alter, gls

SUBFLR
- Doc spl - blk types, tex, old reg
- Rake (Ka) btw blk,relate blks
- (Soil spl on blk top)

CONTACTS
- Trench - sequence
- Dbl core - in youngest

VERY DARK
- Rake
- Doc spl

DARK
- Doc spl

PANS

STA 1

Figure 4.1.1: LMP 31 and CDR-29 tasks in Station 1

places. Two of the speakers, Eugene A. Cernan and Harrison H. Schmitt, were astronauts working directly on the site. Cernan was the Commander (CDR) and Schmitt was the Lunar Module Pilot (LMP). The third speaker, Robert A. Parker, was monitoring the astronauts from Earth. This setting is similar to a human-robot USAR activity, where a robot works together with a rescue team in a disaster site. The robot may have some level of autonomy which allows it to operate on its own, but it is still operated by a human operator working remotely, away from the disaster site. The human operator, therefore, observes the disaster site partially and with uncertainty. Since the Apollo 17 data also shows a partial uncertain observation, it is useful as basis data.

We annotated 1000 utterances from the transcript, particularly 524 utterances from the Geology Station 1 part from minute 122:08:39 to 122:36:49 (i.e. about 28 minutes in duration), and 476 from the Geology Station 2 part between minutes 142:52:53 and 143:17:49 (i.e. about 25 minutes in duration). In total, the annotated data contains 564 turns. The data contains a lot of thematic changes which provide good examples for structuring the mechanisms of thematic change.

Like a human-robot USAR dialogue, the dialogue in the Apollo 17 Journal is a task-oriented dialogue and the tasks in the mission were very well-structured. The CDR and LMP have separate checklists of tasks to accomplish. As shown in Figure 4.1.1[1], CDR and LMP have identical checklists in Geology station 1 and 2. They collaborate and help each other to accomplish their shared-tasks. Similarly, a rescue robot also co-operates with a human operator in accomplishing a USAR task. The task of observing an area or a sample material is similar to the task of a rescue robot to observe a disaster area and the entities within it. However, a robot has limited capabilities compared to a human rescuer. For instance, it might not be designed to perform a physical task such as removing an obstacle, but only to observe and report what it sees.

Both human-robot USAR and the Apollo 17 dialogues are situated dialogues that progress with respect to the changes in dialogue participants' actions, environment and time. The Apollo 17's mission is situated on the moon and the

[1] http://www.hq.nasa.gov/alsj/a17/cuff17.html

astronauts had to observe different places and perform different tasks in different places. As a consequence, their conversation is relative to their location, and the objects and tasks in the location. On the other hand, a USAR task is typically performed in a stressful situation, where one must perform a task in a dangerous place with limited time. A conversation in such a situation must not be verbose, but short and concise. Likewise, the Apollo 17's mission is also constrained by time. The CDR and LMP had limited time to collect samples in one place and then they had to move to another place.

Our approach takes higher level inputs (i.e. at the intentional level, such as speech acts) rather than raw ASR text output. For annotating the dialogue data, we define a set of annotation labels denoting semantic and pragmatic meanings. The annotation labels are described in section 4.1.1. Moreover, since an MLN model takes inputs in first-order logic, we convert the text dialogue data into a logical representation as described in section 4.1.2.

4.1.1 Dialogue Annotation

An utterance is a sequence of words uttered by a speaker at some point in a dialogue. It is a smaller unit than an exchange or a turn and is not necessarily a sentence. A speaker can produce more than one utterance in a turn. As described in section 2.1, an utterance contains a theme which refers to the most important thing in the utterance and a rheme which explains the theme. A dialogue theme represents the salience at a higher level than an utterance. It expresses the most central thing that is focused on at the dialogue level. We attempt to model the theme at the dialogue level, not at the utterance level.

A dialogue theme is not a word or a phrase, but a semantic or pragmatic object being talked about at some point in time during the course of the dialogue. It may refer to an object in the real world which can be either concrete (e.g. scoop, rake, bag) or abstract (e.g. a core at this site). We call such an object Entity. Moreover, people do not only talk about objects, but also about activities or tasks (e.g. get a sample). An activity can be a joint activity (i.e. an activity performed by more than one person) or an individual activity (i.e. an activity performed by a single person). In a task-oriented dialogue, such as our dialogue data, the dialogue participants talk a lot about joint activities or tasks they want to accomplish. They make plans on how they are going to accomplish a task. A human-human conversation typically involves a lot of problem solving and planning. This is not likely to be the case in a human-robot dialogue. Human-robot interaction are simpler than human-human interaction (see section 6.2).

For each utterance, we identify the most important object or/and task being talked about and annotate them as an Entity or a Task respectively. Other objects which are not in the dialogue participants' focus of attention are not annotated. To identify and annotate Entities, we adopt Sidner's focusing algorithm [Sidner, 1979] mentioned in section 2.3.1. By default, the object(s) of an utterance is expected to be a dialogue theme. If a speaker does not use the subject of his utterance to refer to himself and/or to his hearer (e.g I, we, you), the subject is a dialogue theme.

A Task reflects the intention of a speaker, for example what he is going to do or what he wants the hearer to do. Therefore, a dialogue theme may convey a pragmatic meaning with respect to the dialogue context. A Task usually

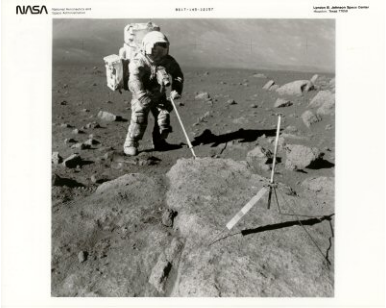


Figure 4.1.2: Harrison H. Schmitt working with a lunar scoop next to a Gnomon on a rock. (NASA photographs)

involves one or more Entities. When a speaker talks about a Task, he may also talk about the Entities involved in the task. In Example (4.1.1),

(4.1.1) Schmitt: Got your gnomon, huh?

the speaker Schmitt asks his hearer whether the hearer has got his gnomon or not. This utterance focuses on the Task "*get a gnomon*" which involves the Entity "*gnomon*" (i.e. a device for calculating the skyline or the altitude of the sun; see Figure 4.1.2[2]). Therefore, both the task and the entity become the dialogue themes of the utterance. Unlike Example (4.1.1), in Example (4.1.2)

(4.1.2) Cernan: Bob, you ready for a mark?

the speaker Cernan asks about the Task "*marking*", (i.e. whether the hearer Bob is ready to perform the task). However, Cernan does not actually talk about the Entity "*mark*".

When a speaker focus his attention to more than one object or task, there is more than a single dialogue theme. These dialogue themes are not a list of ranked Entities/Tasks, but only the most relevant (highly ranked) Entities/Tasks. Sometimes an utterance does not indicate a focus on any Entity or Task, for instance because the speaker did not finish his utterance or due to a technical problem such as an ASR failure during the recognition of the utterance. We annotate the dialogue theme of the utterance as Other. Example (4.1.3) below,

(4.1.3) Schmitt: Gene, do you think...(Pause)

is annotated as Other, because Schmitt did not finish his sentence and the utterance does not focus on any particular Entity or Task. Besides, we use Other to denote all other things that we abstract away from. We show a sample of dialogue annotation of the Apollo 17 data in Table 4.1.

In addition to dialogue themes, we also annotate other cues which are expected to be useful for predicting the dialogue theme in the next utterance. These cues are used to construct features in our MLN (see section 4.2). The Apollo 17 transcripts contain Pauses as in Example (4.1.3). A Pause represents an act of not talking temporarily. As a dialogue is stopped temporarily

[2] http://www.apollomissionphotos.com/reissues/as1714522157r.jpg

Dialogue Text	**Annotation**
Schmitt: Bob, you're going to want a core at this site?	Task: Want a core at this side Entity: A core at this side Question: Want a core at this side
Parker: Roger.	Acknowledgement: Want a core at this side Answer: Want a core at this side
Parker: We'd like to get...	Task: To get
Parker: Number 1 priority will be some block samples, including any dirt that was on the blocks, if there is such.	Entity: Block sample Entity: Priority
Parker: And then the second priority is a rake soil sample;	Entity: Rake sample

Table 4.1: A dialogue annotation sample

in a Pause, the dialogue participants stop talking about the last theme. Moreover, they usually take some time to think, perform actions or carry out a task. Implicitly, the dialogue participants focus their attention to the actions they do in Pause. After a Pause, the dialogue participants are likely to talk about the actions in the Pause or the results of the action. We suppose that these actions and results can be of a different theme from the last theme before the Pause. Therefore, we use Pauses to characterize theme shifts.

Moreover, a speaker always perform a speech act which reflects his/her intention (i.e. what he actually means in his utterance). Speech acts can be useful to characterize thematic changes. In this work, we use three kinds of speech acts, namely Question, Answer and Acknowledgement.

Intuitively, a Question about something can be expected to be followed by an Answer about what has been asked. Besides, from data observation, we found that a Question might also be followed by another Question which still has the same focus\theme as the first Question. Therefore, a Question seems to be a good indicator for a continuation of a dialogue theme. On the other hand, an Answer seems to indicate a shift of the dialogue theme. When a Question is answered, we assume that there is a tendency that an anti-climax occurs in the next utterance. Therefore, the dialogue participants are likely to start talking about a new theme. Moreover, a Question expresses the intention of a speaker to acquire information about some Task or Entity. For instance, the Question in Example (4.1.1) asks information about "*gnomon*", namely whether it has been fetched. A Question may also convey the intention of a speaker to ask a hearer to perform some Task, such as in Example (4.1.2). We annotate a Question or an Answer concerning only a single Task or Entity. If a Task involves some Entity, the Task is preferred in favor of the Entity.

In a conversation, people tend to acknowledge each other's contributions in order to build common ground. Clark and Schaefer [1989] explain that an Acknowledgement is needed to move on in a conversation. As we inspected the dialogue data, we discovered that an Acknowledgement is likely to be followed by a theme shift in the next utterance. Therefore, we take advantage of the occurrence of an ACKNOWLEDGEMENT to predict a new theme. An utterance is annotated as an ACKNOWLEDGEMENT of some TASK or ENTITY, if the utterance expresses a recognition of the TASK or ENTITY. Similar to QUESTIONS, ACKNOWLEDGEMENTS represent intentions of the speaker, such as agreeing on some information about an ENTITY or on performing a proposed TASK. In less frequent cases, some utterances may contain an ACKNOWLEDGEMENT about something OTHER. For example, an ACKNOWLEDGEMENT refers to some TASK which is physically done on the site, but the TASK is not explicitly mentioned in the dialogue. Example (4.1.4) below

(4.1.4) Cernan: (Putting the sample bags in Jack's SCB) Okay.

is an ACKNOWLEDGEMENT with an OTHER theme. An ACKNOWLEDGEMENT is associated only to a single TASK or ENTITY, where a TASK is preferred to an ENTITY.

Label	Total
TASK	94
ENTITY	142
OTHER	113

Table 4.2: The Number of Distinct Dialogue Themes

Table 4.2 shows the number of distinct ENTITIES, TASKS and OTHERS in the data. The table shows that the data contains more distinct ENTITIES than distinct TASKS and a considerable number of distinct OTHERS. Despite the fact that OTHERS may represent different linguistics acts, in this work, we simplify it into a single object. The total number of distinct dialogue themes is smaller than that of utterances, which means that there are repetitions of dialogue themes in the data.

Label	Frequency
QUESTION	82
ANSWER	60
ACKNOWLEDGEMENT	150
PAUSE	83

Table 4.3: Frequency of Speech Act Annotations and PAUSES

Table 4.3 shows the frequency of different speech acts annotated in the data. The data contains more QUESTIONS than ANSWERS, suggesting that not every QUESTION is answered. Besides, the high number of ACKNOWLEDGEMENTS suggests that ACKNOWLEDGEMENTS play a major role in performing a joint activity, and that dialogue participants frequently do grounding to establish common ground. Moreover, there are 83 occurrences of PAUSES in the data illustrating the dialogue participants frequently made a PAUSE in performing their activity.

4.1.2 Logical Forms

A text dialogue has to be transformed into a logical representation before it can be used as an input for an MLN model. We carry out the transformation according to the definitions of formulas in Figure 4.1.3. Each argument of the predicates is a typed variable, which means that it can only be grounded with a constant of its corresponding type. For instance, the arguments of the predicate *utterance* are typed u (utterance-id) and s (speaker's name). For each utterance in a dialogue, we generate ground atoms of the predicate *speaker*, *precede* and *utterance*. The predicate *precede* is defined to specify the order of two utterances, namely that the first argument precedes the second argument. For instance, *precede(U1,U2)* means that the utterance $U1$ precedes the utterance $U2$. We illustrate a sample of the logical representation of the data in 4.4.

Utterance
$speaker(s)$
$precede(u, u)$
$utterance(u, s)$

Dialogue Theme
$entity(e)$
$entityId(e, u)$
$task(a)$
$taskId(a, u)$
$other(o)$
$otherId(o, u)$
$themeIn(u, u)$

Question and Answer
$question(q)$
$questionIn(q, u)$
$questionAbout(q, u)$
$answer(ans)$
$answerIn(ans, u)$
$answerAbout(ans, u)$

Acknowledgement
$ack(ac)$
$ackIn(ac, u)$
$ackOf(ac, u)$

Pause
$pause(p)$
$pauseIn(p, u)$

Figure 4.1.3: MLN Atomic Formulas

As described in the previous section, a dialogue theme can be an Entity, a Task or something Other. The argument e of the predicate *entity* refers to an Entity which gives a description of an object (e.g an entity name) in a domain. Similarly, the arguments a of the predicate *task* and o of the predicate *other*, refer to a Task or an Other object respectively. Each Entity, Task or Other is given a relative identifier (id). The id is relative to the utterance-id where the Entity, Task or Other has been introduced first. We use the notion of relative identifiers to avoid a sparse data problem (e.g. too many entities with a low frequency of occurrence).

The predicate *themeIn* describes the relative location of the dialogue theme (the focus) of an utterance (the first argument). If the dialogue theme of an utterance is new (i.e. it has never been introduced before), the second argument is the same as the first argument. If the dialogue theme has already been introduced before, the second argument is the utterance-id of the utterance where the dialogue theme has been first introduced. For instance, in Table 4.4, entity "*priority*" is a dialogue theme of utterances $U4$ and $U5$. Since "*priority*" appeared first in $U4$, the dialogue theme of $U5$ is referred to $U4$. This can be written as *themeIn(U5,U4)* which is read "the themein $U5$ is in $U4$".

If an utterance contains a Question, we create three ground atoms using the predicates *question*, *questionIn* and *questionAbout*. Firstly, the predicate

Dialogue Text	**Ground Atoms**
Schmitt: Bob, you're going to want a core at this site?	$speaker(Schmitt)$ $utterance(U1, Schmitt)$ $task(Want\_a\_core)$ $taskid(Want\_a\_core, U1)$ $entity(A\_core\_at\_this\_site)$ $entityid(A\_core\_at\_this\_site, U1)$ $question(Q1)$ $questionIn(Q1, U1)$ $questionAbout(Q1, U1)$ $focusIn(U1, U1)$
Parker: Roger.	$speaker(Parker)$ $utterance(U2, Parker)$ $precedeUtt(U1, U2)$ $ack(Ack1)$ $ackIn(Ack1, U2)$ $ackOf(Ack1, U1)$ $answer(Ans1)$ $answerIn(Ans1, U2)$ $answerAbout(Ans1, U1)$ $focusIn(U2, U1)$
Parker: We'd like to get...	$utterance(U3, Parker)$ $precedeUtt(U2, U3)$ $task(To\_get)$ $taskid(To\_get, U3)$ $focusIn(U3, U3)$
Parker: Number 1 priority will be some block samples, including any dirt that was on the blocks, if there is such.	$utterance(U4, Parker)$ $precedeUtt(U3, U4)$ $entity(Block\_sample)$ $entityid(Block\_sample, U4)$ $entity(Priority)$ $entityid(Priority, U4)$ $focusIn(U4, U3)$ $focusIn(U4, U4)$
Parker: And then the second priority is a rake soil sample;	$utterance(U5, Parker)$ $precedeUtt(U4, U5)$ $entity(Rake\_sample)$ $entityid(Rake\_sample, U5)$ $focusIn(U5, U3)$ $focusIn(U5, U4)$ $focusIn(U5, U5)$

Table 4.4: Logical Annotation of a Sub-dialogue

question defines the question-id. Secondly, the location where the QUESTION appears (i.e. an utterance's id) is described by the predicate *questionIn*. Finally, the predicate *questionAbout* describes the location of what the QUESTION is about, that is, the utterance id where the ENTITY, TASK or OTHER in question was introduced. Similar to *themeIn*, *questionAbout* contains the relative location of the dialogue theme in the QUESTION. For example in Table 4.4, utterance $U1$ contains a QUESTION defined by *question(Q1)* and *questionIn(Q1,U1)*. Since the QUESTION is about the task *Want_a_core* introduced in $U1$, we create *questionAbout(Q1,U1)*.

Similar to a QUESTION, for an utterance containing an ANSWER, three ground atoms are created using the predicate *answer*, *answerIn*, and *answerAbout*. The predicate *answer* defines the answer-id and the predicate *answerIn* describes where the ANSWER appears (i.e. the current utterance's id). Additionally, the predicate *answerAbout* describes the location of what the ANSWER is about. For example, in 4.4 utterance $U2$ contains an ANSWER about a dialogue theme in $U1$. Therefore, we created the following ground atoms: *answer(Ans1), answerIn(Ans1,U2)* and *answerabout(Ans1,U1)*.

An ACKNOWLEDGEMENT also has three ground atoms similar to those of a QUESTION or an ANSWER. The predicate *ack* defines the acknowledgement-id, the predicate *ackIn* describes where the ACKNOWLEDGEMENT appears (i.e. the current utterance's id), and *ackOf* describes the location of what is being acknowledged. An utterance can express both an ACKNOWLEDGEMENT and an ANSWER. For example in utterance $U2$ in Table 4.4, Parker acknowledges Schmitt's proposal and at the same time gives his positive answer to Schmitt's question.

Finally, for every occurrence of a PAUSE such as the one in Example (4.1.3), two ground atoms are created. The predicate *pause* defines the pause-id and *pauseIn* describes where the PAUSE has occurred.

4.2 MLN Model

We use the Alchemy tool[3] [Kok et al., 2009] to define a set of first-order logic formulas for constructing an MLN model. The formulas are described in section 4.2.1. We also use the tool for learning the weights of each formula, and for testing an MLN model in predicting the next theme. In section 4.2.2, we describe the inputs of an MLN model for learning weights and the queries for predicting the next theme. In section 4.2.3, we describe the outputs (predictions) of an MLN model.

4.2.1 MLN First Order Logic Formulas

Following the notation in the Alchemy tool, we use a shortened notation for defining first-order logic formulas. All arguments of the first order logic formulas are variables written in small letters. Whenever free variables occur in the antecedent (left side of the arrow), there is an implicit universal quantifier of each of the free variables around the whole formula.

First of all, atomic formulas are defined as shown in Figure 4.1.3. We also specify the uniqueness property of each ENTITY, TASK and OTHER in Figure

[3] http://alchemy.cs.washington.edu/

Entity
$\forall e1\ (entity(e1) \rightarrow \exists fid1\ entityid(e1, fid1))$.
$entity(e1) \wedge entityid(e1, fid1) \wedge entityid(e1, fid2) \rightarrow fid1 = fid2$.
$entity(e1) \wedge entityid(e1, fid1) \wedge entity(e2) \wedge entityid(e2, fid1) \rightarrow e1 = e2$.

Task
$\forall a1\ (task(a1) \rightarrow \exists fid1\ taskid(a1, fid1))$.
$task(a1) \wedge taskid(a1, fid1) \wedge taskid(a1, fid2) \rightarrow fid1 = fid2$.
$task(a1) \wedge taskid(a1, fid1) \wedge task(a2) \wedge taskid(a2, fid1) \rightarrow a1 = a2$.

Other
$\forall o1\ (other(o1) \rightarrow \exists fid1\ otherId(o1, fid1))$.
$other(o1) \wedge otherId(o1, fid1) \wedge otherId(o1, fid2) \rightarrow fid1 = fid2$.
$other(o1) \wedge otherId(o1, fid1) \wedge other(o2) \wedge otherId(o2, fid1) \rightarrow o1 = o2$.

Figure 4.2.1: Uniqueness property of an ENTITY, a TASK and an OTHER

4.2.1. The rules specify that every ENTITY, TASK or OTHER is unique (i.e. has a unique description) and has a unique identifier, which is relative to the utterance-id where it is introduced. Each of these rules is marked with a full-stop. That means these rules are hard-constrained and not associated with any weight.

In this work, an MLN model predicts a relative dialogue theme, that is whether a dialogue theme of the current utterance is new or identical to a previous utterance. We do not specifically predict an appearance of an ENTITY, a TASK or an OTHER, because predicting an absolute identifier will lead to a sparse data problem. Therefore, an MLN model predicts whether a theme continues from the last utterance (U_{n-1}) or changes in the current utterance (U_n). Formula 4.2.1 where $precede(u1, u2) \Rightarrow u1! = u2.$, predicts that a theme from U_{n-1} is the theme in U_n. Formula 4.2.2 predicts a new theme (*Shift New*) in U_n regardless of the theme(s) in U_{n-1}. To penalize the probability of other themes, we define formula 4.2.3 which specifies that, if there is a new theme in an utterance, other themes are not the theme of that utterance. For smoothing, we define formula 4.2.4 which says that a theme in an utterance can be a theme in another utterance regardless of the order of the utterances.

$$precede(u1, u2) \wedge themeIn(u1, fid1) \rightarrow themeIn(u2, fid1) \qquad (4.2.1)$$

$$precede(u1, u2) \wedge themeIn(u1, id1) \rightarrow themeIn(u2, u2) \qquad (4.2.2)$$

$$themeIn(u2, u2) \wedge fid1! = u2 \rightarrow !themeIn(u2, fid1) \qquad (4.2.3)$$

$$themeIn(u1, id1) \rightarrow themeIn(u2, id1) \qquad (4.2.4)$$

A dialogue theme of type OTHER is complex because it includes different linguistic acts and because the reaction of a dialogue participant to an OTHER theme may vary. Different strategies can be used to specifically model human behavior in performing a linguistic act and giving response to it. In this work, we handle an OTHER theme by simply predicting a new theme after an appearance of an OTHER theme in the last utterance (see Formula 4.2.5).

Question and Answer
$questionIn(q1, u1) \rightarrow question(q1).$
$answerIn(ans1w, u1) \rightarrow \exists answ\ (answer(answ)).$
$answerAbout(answ, u1) \rightarrow \exists answ\ (answer(answ)).$
$precedeUtt(u1, u2) \wedge questionIn(q1, u1) \rightarrow \exists answ\ (answerIn(answ, u2))$

Acknowledgement
$ackIn(ackn, u1) \rightarrow \exists ack1\ (ack(ackn)).$
$ackOf(ackn, fid1) \rightarrow \exists ack1\ (ack(ackn)).$

Pause
$pauseIn(p1, u1) \Rightarrow pause(p1).$

Figure 4.2.2: Properties of a Question, an Answer, an Acknowledgement, and a Pause

$$otherId(o1, oid1) \wedge precede(u1, u2) \wedge themeIn(u1, oid1) \rightarrow themeIn(u2, u2) \quad (4.2.5)$$

We define the properties of a QUESTION and an ANSWER in Figure 4.2.2. By using Formula 4.2.6, an MLN model learns how likely it is that a QUESTION is immediately followed by an ANSWER. Moreover, it takes advantage of the existence of a QUESTION or an ANSWER for predicting the theme. When a speaker asks about a theme, he is likely to continue talking about that theme. Otherwise, it is likely that an ANSWER for that QUESTION will be given. Formula 4.2.7 models this by predicting that U_n has the same theme as QUESTION U_{n-1}. On the contrary, a theme is intuitively not likely to be continued after an ANSWER. Since the theme of the QUESTION has been clarified by an ANSWER, the theme in the next utterance is likely to change. Formula 4.2.8 predicts a new theme in U_n, after an appearance of an ANSWER in U_{n-1}.

$$precede(u1, u2) \wedge questionIn(q1, u1) \rightarrow answ(answerIn(answ, u2)) \quad (4.2.6)$$

$$precede(u1, u2) \wedge questionIn(q1, u1) \wedge questionAbout(q1, fid1) \rightarrow themeIn(u2, fid1) \quad (4.2.7)$$

$$precede(u1, u2) \wedge answerIn(ans1, u1) \rightarrow themeIn(u2, u2) \quad (4.2.8)$$

We also specify the properties of an ACKNOWLEDGEMENT and a PAUSE in Figure 4.2.2. Formula 4.2.9 predicts a new theme after an appearance of an ACKNOWLEDGEMENT. A PAUSE can be a good indication for a thematic change in U_n. Formula 4.2.10 predicts a new theme after a PAUSE.

$$precede(u1, u2) \wedge ackIn(ack1, u1) \rightarrow themeIn(u2, u2) \quad (4.2.9)$$

$$precede(u1, u2) \wedge pause(p1) \wedge pauseIn(p1, u1) \rightarrow themeIn(u2, u2) \quad (4.2.10)$$

Each of the first order logic formula is assigned a weight which roughly indicates the tendency of a formula to be true in a training database described in the next section. The more a formula is inconsistent with the Herbrand interpretation in the training database, the lower is the weight that it gets.

4.2.2 MLN Inputs

We designed first-order logic formulas with the purpose of predicting the dialogue theme in the next utterance. These formulas are associated with weights representing the tendency of each formula to be true. To learn weights and perform inference, an MLN takes inputs written in first-order logic. The weights are learned from a database of ground atoms which describes the possible interpretations of each atomic formula. The database should explicitly specify truth values of all possible ground atoms. Therefore, it should contain not only the positive ground atoms, but also the negative ones.

Our database was built from a text dialogue which is transformed into a collection of grounded atomic formulas (see section 4.1.2). The transformed text dialogue contains only positive ground atoms. For each predicate, we generated all the remaining possible groundings as negative evidence. For example, the positive and the generated negative evidence of the predicate *utterance* in the sub-dialogue of five utterances in Table 4.4, is shown in Table 4.5.

Positive Evidence	Negative Evidence
$utterance(U1, Schmitt)$	$!utterance(U1, Parker)$
$utterance(U2, Parker)$	$!utterance(U2, Schmitt)$
$utterance(U3, Parker)$	$!utterance(U3, Schmitt)$
$utterance(U4, Parker)$	$!utterance(U4, Schmitt)$
$utterance(U5, Parker)$	$!utterance(U5, Schmitt)$
	$!utterance(U1, Cernan)$
	$!utterance(U2, Cernan)$
	$!utterance(U3, Cernan)$
	$!utterance(U4, Cernan)$
	$!utterance(U5, Cernan)$

Table 4.5: Evidence for the predicate *utterance* in Table 4.4

MLNs perform inference to answer a given query. For the purpose of predicting the theme(s) in the next utterance, we query the predicate *themeIn* and we also give some context to it. A context for a query includes:

- the definition of the next utterance
- a dialogue history containing the formal descriptions of one or more preceding utterances
- a re-introduction of some constants in the dialogue history, and
- several free ground atoms.

The size of a dialogue history is the number of previous utterances included in the dialogue history. Using a different dialogue history length may lead to

	Ground Atoms

Dialogue History	*utterance(U699,Parker)*
	ack(Ack104)
	ackIn(Ack104,U699)
	ackOf(Ack104,U11)
	themeIn(U699,U11)

	utterance(U700,Schmitt)
	precede(U699,U700)
	question(Q64)
	questionIn(Q64,U700)
	questionAbout(Q64,U7)
	themeIn(U700,U7)

Next utterance	*utterance(U701,Cernan)*
	precede(U700,U701)

Re-introduction of Constants	*utterance(U11,Schmitt)*
	speaker(Schmitt)
	utterance(U7,Parker)
	speaker(Parker)
	speaker(Cernan)

Free Ground Atoms	*task(Free_ Task)*
	entity(Free_Entity)
	other(Free_ Other)
	question(Free_ Q)
	answer(Free_Answer)
	ack(Free_Ack)
	pause(Free_Pause)

Query	*themeIn*

Table 4.6: A sample query with a dialogue history of size two

different results of an MLN inference. To examine these effects, we provide a query with different lengths of dialogue history. We experimented with dialogue histories consisting of either only the one, two, four, or six last utterances. If an utterance in a dialogue history contains a constant that is introduced outside of the dialogue history, that constant is re-introduced. If a dialogue history does not include the introduction of a theme-id (i.e. the utterance-id where an Entity, Task, or Other is introduced), the theme-id is re-introduced.

To perform inference, all variables specified in an MLN must be grounded in a given dialogue history. In fact, some variables may not be grounded in a dialogue history (i.e. there is no ground atom for some variables in the dialogue history), because the history is a rather short sub-dialogue. Therefore, we add some ground atoms to ground such missing variables. We call these ground atoms *free ground atoms.*

Table 4.6 shows a sample of a query and its context. By using this sample, we would like to predict the dialogue theme(s) in utterance *U701* (i.e. the next utterance). The query context contains a dialogue history of length two which lists all the ground atoms in the utterances *U699* and *U700.* There is an Acknowledgement in the utterance *U699* and a Question in the utterance *U700.* The Acknowledgement is about a dialogue theme (i.e. an Entity, TASK or OTHER) introduced in utterance *U11* and the QUESTION is about a theme introduced in utterance *U7*. Since *U11* and *U7* are not introduced in the dialogue history, they are re-introduced. Similarly, the constants *Parker*, *Schmitt* and *Cernan* are also re-introduced. The theme in *U699* is identical to a theme in *U11* and the theme in *U700* is identical to a theme in *U7*. The free ground atoms are added to anticipate missing ENTITY, TASK, OTHER, QUESTION, ANSWER, ACKNOWLEDGEMENT and PAUSE constants. In this sample, the dialogue history lacks an ANSWER and a PAUSE.

A context only describes positive ground atoms. All other possible groundings of the predicates in the context are implicitly not true. We explicitly specify this by adding negative ground atoms. These negative ground atoms prevent an MLN from predicting probabilities for all implicitly wrong groundings.

4.2.3 MLN Outputs

An MLN model predicts the next theme by performing inference. Particularly, it infers an answer to the query *themeIn*. An MLN model will compute the probability of each possible grounding of the predicate *themeIn*. Since we have generated all the negative evidence of the predicate *themeIn* in a query, an MLN model will compute the possible groundings of the *themeIn* predicate of the next utterance only. The outputs of the MLN inference show how probable it is for each possible theme to appear in the next utterance. The probabilities reflect how an MLN predicts and what its predictions are. We consider the predicate *themeIn* with the highest probability as the MLN theme prediction. In addition to that, we also consider all other themes whose probabilities have low difference from the highest probability.

The probabilities of all possible groundings in MLN outputs do not sum up to 1, and therefore they do not form a probability distribution. In reality, it is possible for an utterance to contain more than one theme. Thus, each possible grounding could actually be a theme in the next utterance. An MLN model reflects this by computing probabilities for all possible groundings that could appear in the next utterance

Possible Grounding	Probability
themeIn(U701,U11)	0.488
themeIn(U701,U7)	0.924
themeIn(U701,U701)	0.646

Table 4.7: Sample results of the query in Table 4.6

Table 4.7 presents sample outputs of an MLN given the query in Table 4.6. For every theme in the dialogue history, the MLN model computes the probability of it becoming the theme of *U701*. The MLN model computes the probability that *U11* (i.e. the theme-id of *U699*) and *U7* (i.e the theme-id of *U700*) will be the theme-id in *U701*. An MLN model does not only consider the theme(s) of the last utterance but also the theme(s) of all other utterances in the dialogue history. In addition to that, an MLN model computes the probability of a new theme.

The probability of a possible grounding shows some degree of certainty (i.e. how certain an MLN model that the grounding is true). The MLN expects with 49% probability that a dialogue theme in *U11* is a theme in *U701*. Moreover, it is 64% sure that there will be a new theme in *U701*. With 92% probability, the MLN claims that the theme in *U7* is/are the theme in *U701*. Since we consider the possible grounding with the highest probability as the MLN theme prediction, the MLN predicts that the theme of *U701* is the same as the theme in *U7*. In this case, the MLN predicts that there will be a continuation of theme in *U701*, because *U7* is the theme-id of *U700*. The MLN has learned that a continuation of theme is likely to happen after an occurrence of a QUESTION. Since *U700* contains a QUESTION, the MLN is pretty certain that there will be a continuation of the theme.

Chapter 5

Evaluation

We carried out experiments using the first-order logic formulas described in the previous chapter. The weights of each formula were learned using different training data sets. We performed 5-fold cross validation to estimate the average performance of the MLN models in practice. The models were evaluated using the annotated data described in section 4.1. Furthermore, we compared the MLN models to two baselines. The first baseline is a random choice from all possible themes within a given dialogue history. The second baseline is a basic theme prediction model which simply continues every theme in the last utterance.

In the next section, we describe the evaluation methods we used and the methodology of the evaluation. Section 5.2 explains how we split the dialogue data to create training and test data sets. Section 5.3 compares the weights of different MLN models learned using different training sets. Finally, we present and analyze the evaluation results in section 5.4.

5.1 Evaluation Methodology

As described in section 4.1, we annotated 1000 utterances from the Apollo 17 Journal. We split the utterances to create a training data set and a test data set. The training data set is used to learn the weights of an MLN model. Different training data sets will result in different weights of the same first-order logic formulas, and therefore they will produce different MLN models (i.e. a set of first-order logic formulas and their weights). An MLN model can be used to predict the focus or foci of each utterance in a test data set by inferring a query. Subsequently, the MLN predictions (i.e. the outputs of MLN inference) can be evaluated by comparing them to the theme-annotations in the test data set.

Our dialogue data size is fairly small, and therefore it might not be representative of real data in practice. Using a small training data set may lead to an over-fitting problem. In our case, for instance, a resulting MLN model may fit the training data set very well, but still not fit the test data. This can happen because the MLN model learns the characteristics of thematic changes from the training data set. The characteristics in the training data set, however, may be different from the thematic changes in the test data.

Cross validation is a technique to estimate the performance of a model across multiple training and test data sets, so that the model performance is not biased by certain training and test data sets. In k-fold cross validation, we run k-rounds of training and testing using different training and test data sets. First of all, the data is partitioned into k-subsamples. Each subsample is then used as a test data set for one round of cross-validation. Therefore, we will have k test data sets. The k-th test data set is the test data set for the k-th round of cross-validation. For each round, the remaining k-1 subsamples are used to create a training data set. Each training data set shares a subset of another training data set, but in principle it should still be different from each of other training data sets.

Section 5.2 explains how we split our data to perform 5-fold cross-validation. To obtain representative weights, an MLN needs an adequate amount of training data. On the other hand, we also need enough data for testing and evaluating the MLN model's performance. We use two sizes of training data sets, namely sets of 500 and 700 utterances in size, to measure the influence of the size in learning weights and model performance. The rest of the data is used to create a test data set.

For each training data size, we perform 5-fold cross validation. In each round of cross validation, an MLN model is generated. Each 5-fold cross validation will produce five MLN models. Because we use two training data sizes, we have 10 MLN models (i.e. 2 training data sizes × 5-rounds of cross validation). We compare the weights of the MLN models in section 5.3 and we show that the standard deviations of the weights of the first-order logic formulas are generally low. This suggest that the weights obtained from different rounds of cross validation are fairly close to the expected weight (i.e. the mean of the weights).

Furthermore, as explained in section 4.2.2, we are interested in observing the effects of varying the dialogue history size. The size of a dialogue history is the number of utterances it contains. In our experiments, we used dialogue history lengths one, two, four and six. We performed 5-fold cross validation for each dialogue history size. In total, we ran 40 experiments (i.e. 2 training data sizes × 5-rounds cross validation × 4 dialogue history sizes) using 10 MLN models.

We evaluated MLN outputs by comparing them to the annotation of their corresponding test data set. For each experiment, we computed the average precision, recall and F1 score of the MLN model over all utterances in a test data set. The precision of theme prediction of an utterance is computed as the number of predictions which are identical to the annotated themes of the utterance, divided by the number of all MLN predictions for the utterance.

$$Precision = \frac{the\ number\ of\ predictions\ identical\ to\ the\ annotations}{the\ number\ of\ predictions}$$

The second evaluation measure we use is recall. It is defined as the number of predictions that are identical to the annotated themes, divided by the number of the annotated themes for the utterance.

$$Recall = \frac{the\ number\ of\ predictions\ identical\ to\ the\ annotations}{the\ number\ of\ annotations}$$

The F1 score combines precision and recall with equal importance.

$$F1 = 2 * \frac{Precision * Recall}{Precision + Recall}$$

For each utterance, we selected the MLN prediction that has the highest probability. Additionally, we selected all other predictions whose probabilities are less than 0.05 below the highest probability.

We compared the average precision, recall and F1 scores of the MLN models that we obtained using different training sizes. We also compared the models to two baselines, a random baseline and an informed baseline. The random baseline predicts the next theme by randomly picking one of the themes in the dialogue history of given length, or a new theme. On the other hand, the informed baseline is a basic theme prediction model which simply continues all the themes of the last utterance. For the sample query and its context in Table 4.6, the informed baseline will predict *themeIn(U701,U7),* since *U7* is the theme in *U700.* The informed baseline is adapted from centering theory [Grosz et al., 1995] described in section 2.3.4. The informed baseline is slightly different from centering theory in the way that it continues every theme in the last utterance, while centering theory only continues a single theme of the last utterance.

Furthermore, we investigate the performance of the MLN models in predicting different types of thematic/focus change. As described in section 1.2, we examine three types of thematic changes, namely *Continuation, Shift New* and *Shift Old.* For *Shift Old*, we consider the themes of the utterances in a dialogue history, apart from the last utterance. We computed how many times the MLN models successfully predict the annotated theme of each utterance in each test data set. A prediction is successful if it is identical to the theme annotation. In other words, we compute the recall for each thematic type. For the recall of *Continuation*, we computed the number of predicted *Continuations* identical to the annotation. We calculated the recall of *Shift New* and that of *Shift Old* in a similar way. We describe the results of this investigation in section 5.4.4.

Finally, we analyzed the errors produced by the MLN models in predicting the next theme. An error is a failure in predicting the next theme precisely as the theme in the consulted annotation. We observe three kinds of errors:

- wrong prediction: predicting a different theme from the annotation
- lack of prediction: not predicting a theme in the annotation
- over-prediction: predicting an additional theme other than the themes in the annotation

The error analysis is presented in section 5.4.5.

5.2 Data Splitting

Our annotated dialogue data consists of 1000 utterances. We numbered the utterances sequentially from 1 to 1000. We used 500 and 700 sequential utterances to create two training data set sizes. We call the respective training data sets TRAIN500 and TRAIN700. Moreover, we used test data sets of 300 sequential utterances which are neither contained in the TRAIN500 nor in the

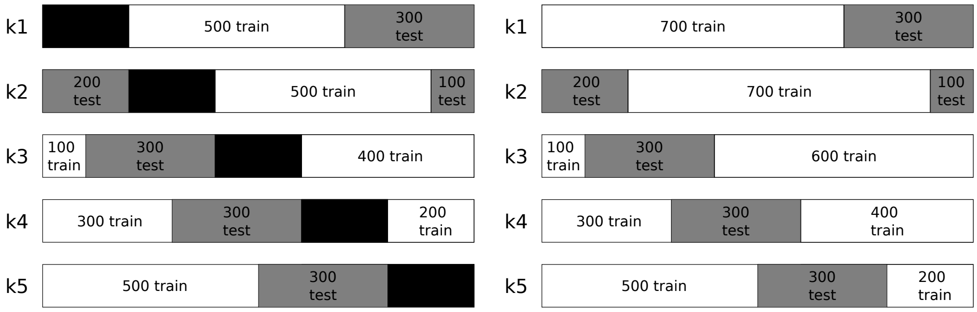


Figure 5.2.1: Data splitting over 5-fold cross validation

TRAIN700 data sets. An MLN model trained with a TRAIN500 set is called an MLN500 model, and an MLN model trained with a TRAIN700 set is called an MLN700 model. Different training sets produce different weights for the same first-order logic formulas and therefore they yield different MLN models.

We used cross validation to estimate the performance of the MLN500 and MLN700 models over a variety of data. We chose 5-fold cross validation ($k1 - k5$) using different combinations of the training and test data sets. Our training and test data set selections are visualized by Figure 5.2.1. The left side of the Figure shows the data splitting for 5-fold cross validation using TRAIN500 sets, whereas the right side shows the data splitting for 5-fold cross validation using TRAIN700 sets. The test data sets are exactly the same in both sides.

Since dialogue data is one-dimensional and sequential, we cannot choose utterances randomly and collect them as a training or a test data set. Instead, we take blocks of utterances for our training and test data sets. We split our dialogue data into three blocks of utterances:

- a training data set (white block in Figure 5.2.1),
- a test data set (gray block in Figure 5.2.1), and
- the rest of the utterances which are not used (black block in Figure 5.2.1).

We made variations of the training and test data sets by shifting the blocks. A TRAIN700 set always contains a corresponding TRAIN500 set and also contains additional 200 utterances. These 200 utterances comprise the black boxes in the left side. To avoid over-fitting the TRAIN500 sets to certain parts of the data, the black blocks are gradually shifted over all cross-validations that together cover all utterances with the same frequency. This is important because if, for example, the black blocks only stayed at the beginning and at the end of the data, the TRAIN500 sets would be biased towards the middle part of the data. That means that the models would learn much of the characteristics of the thematic changes in the middle part, but they would miss the characteristics in the other parts. If the characteristics in these other parts are different from those in the middle part, the trained models might fail to predict those thematic changes in the test data sets that have such characteristics.

Table 5.1 summarizes the different sets of utterances for the TRAIN500, the TRAIN700 and the test data sets. The TRAIN500 set for the first round of cross validation $k1$ is a sub-dialogue containing all utterances from utterance

K	TRAIN500 set	TRAIN700 set	Test set
1	201-700	1-700	701-100
2	401-900	201-900	1-200 and 900-1000
3	1-100 and 601-1000	1-100 and 401-1000	101-400
4	1-300 and 801-1000	1-300 and 601-1000	301-600
5	1-500	1-500 and 801-1000	501-800

Table 5.1: The training and test data sets for performing 5-fold cross validation (in utterance number-id)

number 201 to 700. The TRAIN700 set for $k1$ contains all utterances from utterance number 1 to 700, which means that it covers the TRAIN500 set. The test data for round $k1$ contains the remaining utterances from utterance number 701 to 1000. For round $k2$, the TRAIN500 set contains all utterances from 401 to 900, and the TRAIN700 set from 201 to 900. The test data set for round $k2$ combines the first 200 utterances at the beginning of the dialogue data and the last 100 utterances at the end of the data. For round $k3$, the TRAIN500 set combines two sub-dialogues, namely the first 100 utterances at the beginning of the data and 400 utterances at the end of the data. Similar to the TRAIN500 set, the TRAIN700 set for round $k3$ uses the first 100 utterances at the beginning of the data and the last 600 utterances at the end of the data. The TRAIN500 set for round $k4$ combines the first 300 utterances at the beginning of the data and 200 utterances at the end of the data. The TRAIN700 set also contains the first 300 utterances and 400 last utterances. The TRAIN500 set for the last round $k5$ uses the first 500 utterances. The TRAIN700 set includes the TRAIN500 and also the 200 last utterances at the end of the data.

5.3 MLN Weights

An MLN learns weights for all its first-order logic formulas that are not hard-constrained (i.e. those that do not end with a full-stop). A weight of a formula does not designate the probability of the formula, but represents the tendency of the formula to be satisfied in a given training data set. The higher the weight of a formula, the more likely it is that the formula is true in a data set. In contrast, the lower the weight of a formula, the more unlikely it is that the formula is true in a data set. The characteristics of different training sets can be different from one another. Some formulas may have higher tendencies in some training sets and lower tendencies in other training sets.

We used the different training sets described in section 5.2 to learn weights for the first-order logic formulas described in 4.2.1. In Tables 5.2, we summarize the weights of Formula 4.2.1 through Formula 4.2.10 that were learned using the TRAIN500 sets over 5-fold cross validation. Descriptions of the formulas are given in Figure 5.3.1. For each formula, we computed the means and standard deviations of all the weights learned over 5-fold cross validation (i.e. of all MLN500 models). The mean of a formula shows the average tendency of the formula to be true. The standard deviation of a formula indicates how far its weights in different rounds of the cross validation deviate from the average weights of the formula. The higher the standard deviation, the larger the

FOL	5-Fold Cross Validation					Mean	Std. Dev
Formula	k1	k2	k3	k4	k5		
4.2.1	2.483	2.752	2.251	0.453	2.772	2.142	0.986
4.2.2	1.653	0.248	1.583	0.329	1.730	1.109	0.751
4.2.3	0.005	-0.010	0.002	0.000	0.009	0.001	0.007
4.2.4	0.007	0.008	0.007	0.043	0.017	0.017	0.016
4.2.5	0.141	1.529	-0.060	0.103	-0.175	0.307	0.694
4.2.6	1.577	2.923	1.499	0.348	1.860	1.641	0.920
4.2.7	4.315	4.798	3.875	0.824	4.696	3.702	1.649
4.2.8	3.096	1.657	3.097	0.439	2.775	2.213	1.155
4.2.9	3.080	1.709	3.070	0.448	2.815	2.225	1.142
4.2.10	0.088	-1.156	-0.417	0.224	-0.278	-0.308	0.541

Table 5.2: The weights of the first-order logic (FOL) formulas learned using TRAIN500s over 5-fold cross validation

Formula	**Description**
4.2.1	Continue the theme of the last utterance
4.2.2	Predict a new theme
4.2.3	No other theme when there is a new theme
4.2.4	A theme is a theme in another utterance
4.2.5	Predict a new theme after an other
4.2.6	Predict an answer after a question
4.2.7	Continue of the theme of a question
4.2.8	Predict a new theme after an answer
4.2.9	Predict a new theme after an acknowledgement
4.2.10	Predict a new theme after a pause

Figure 5.3.1: Description of first-order logic formulas of the MLN models

deviations are. The average standard deviation over all the first-order logic formulas is 0.784, which is fairly low. This suggests that the weight of each formula does not deviate too much from its expected weight (i.e. the mean of the weights obtained from 5-fold cross validation).

Figure 5.3.2 highlights the changes of the weights of each first-order formula over 5-fold cross validation. Generally, the weights of all formulas in $k4$ get lower than in other k. We suspect that this is because the characteristics of the thematic changes in the training data have a wide range of variation and many are not explained by the first-order formulas. So, the formulas are often inconsistent with the annotated training data. The formulas receive low weights due to the high rate of inconsistency. Moreover, the characteristics are almost equally distributed and there are no characteristics that particularly stand out. Therefore, the weights of the formulas are quite similar and fairly low.

Despite the low weights, in section 5.4.2 we show that the average performance of MLN500 models using $k4$ is better than the average performance over all training sets. This suggests that the higher weights of first-order logic formulas do not necessarily improve an MLN model performance. The weights can be interpreted as the strength of the tendency of the first-order logic formula to be true. However, the degree of strongness does not explicitly explain the

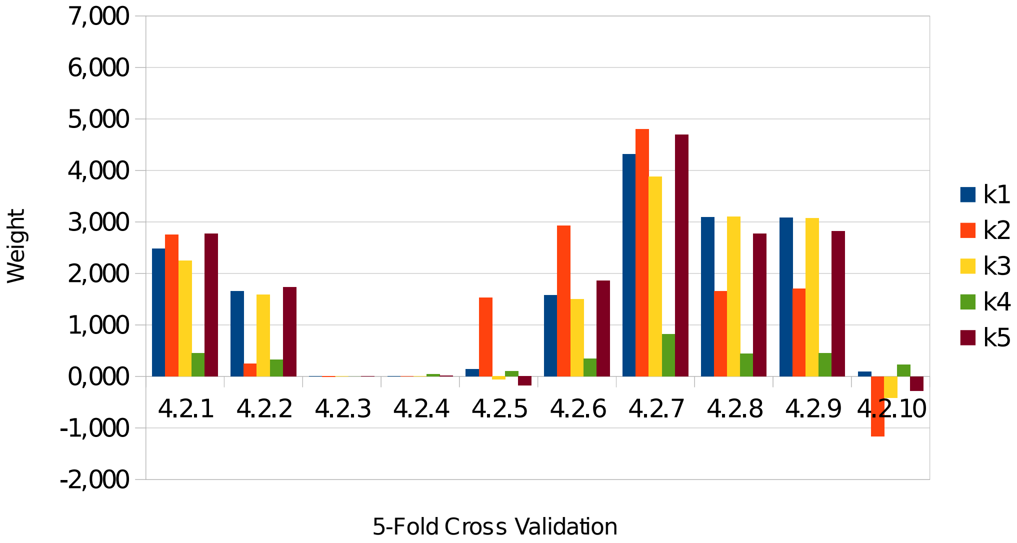


Figure 5.3.2: The fluctuation of the MLN500 models' weights over 5-fold cross validation

performance of an MLN model. Moreover, a weight should not be interpreted from the value itself, but from its relations to other weights.

Formula 4.2.1, which predicts a continuation of the theme of the last utterance, is much less likely to be true in $k4$ compared to the other rounds of cross validation. On the other hand, Formula 4.2.4, which specifies that a theme in an utterance can also be a theme in another utterance, and Formula 4.2.10 which predicts a new theme after an appearance of a PAUSE in the last utterance, get higher weights in $k4$. This suggests that there is more evidence in $k4$ that increases the tendency of Formula 4.2.4 and Formula 4.2.10 to be true.

However, Formula 4.2.4 is generally not likely to be true (i.e. a theme in an utterance does not tend to be a theme in another utterance). Although a theme may be a theme of another utterance, no theme is likely to be the theme of every other utterance. We believe that the formula received a low weight for that reason. Over all five rounds of cross validation, the MLN models also learned that the Formula 4.2.10 is rather unlikely, except in $k4$. In other words, according to the training data sets, a PAUSE does not really tend to be followed by a new theme.

The prediction of a new theme, which is crafted in Formula 4.2.2, is less likely to be true in $k2$ and $k4$, but more likely in $k1$, $k3$ and $k5$. Overall, the MLN models tend to predict a continuation of theme rather than a new theme. The MLN models also believe that a new theme does not tend to be a single theme of an utterance, as indicated by the low weights of Formula 4.2.3 over all k. The tendency of a new theme to occur after an OTHER theme (i.e. the Formula 4.2.5), is generally low but higher than that of Formula 4.2.3.

On the other hand, the weights of Formula 4.2.6 show that a QUESTION has a slight tendency to be followed by an ANSWER. Moreover, Formula 4.2.7 is assigned the highest average weight. This means that the theme in a QUESTION is very likely to be continued in the next utterance. Moreover, the weights of

FOL	5-fold Cross Validation					Mean	Std. Dev
Formula	k1	k2	k3	k4	k5		
4.2.1	0.444	0.465	2.920	3.500	0.927	1.651	1.450
4.2.2	0.333	0.382	2.056	2.263	0.552	1.117	0.958
4.2.3	0.000	0.000	0.005	0.029	0.002	0.007	0.012
4.2.4	0.037	0.026	0.006	0.004	0.025	0.020	0.014
4.2.5	0.107	0.201	-0.050	-0.102	2.101	0.452	0.930
4.2.6	0.378	0.400	1.864	2.926	0.489	1.211	1.145
4.2.7	0.834	0.889	5.026	6.422	1.526	2.940	2.603
4.2.8	0.404	0.448	3.679	3.121	1.126	1.756	1.541
4.2.9	0.426	0.476	3.662	3.181	1.098	1.768	1.541
4.2.10	-0.037	0.116	0.198	0.684	0.610	0.314	0.316

Table 5.3: The weights of the first order logic formulas learned using TRAIN700s over 5-fold cross validation

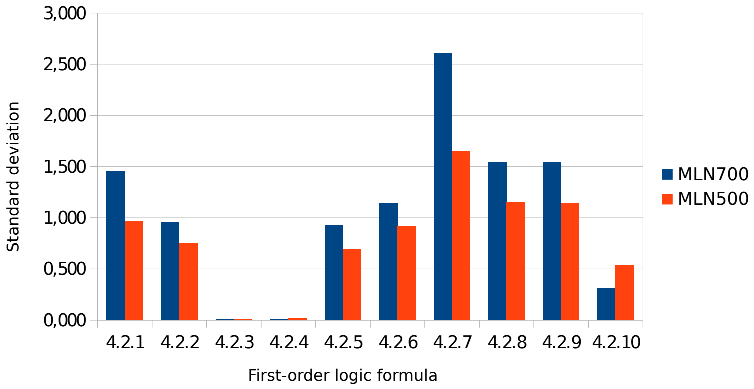


Figure 5.3.3: Standard deviations of the weights of the MLN500 and MLN700 models

Formula 4.2.8 show that a new theme is likely to occur after an occurrence of an Answer in the last utterance. It is similar to the tendency of an occurrence of a new theme after an Acknowledgement as suggested by the weights of Formula 4.2.9.

Table 5.3 is similar to Table 5.2. It shows the weights of the first order logic formulas obtained using TRAIN700 sets over 5-fold cross validation. The standard deviations of the weights of the MLN700 over 5-fold cross validation have a similar trend to those of MLN500 weights. The average standard deviation of the MLN700 weights, however, is slightly higher than that of the MLN500 weights. The comparison between the standard deviations of the weights of the MLN500 and MLN700 models is illustrated in Figure 5.3.3. The Figure shows that weights of Formula 4.2.3 and Formula 4.2.4 in both MLN500 and MLN700 models are very stable over 5-fold cross validation. On the contrary, there is a large deviation in the distribution of the weights of Formula 4.2.7. The deviations of the weights of the other formulas are quite normal because they only vary approximately one unit from the their expected weights.

The characteristics of thematic/focus changes in the TRAIN700 sets can be

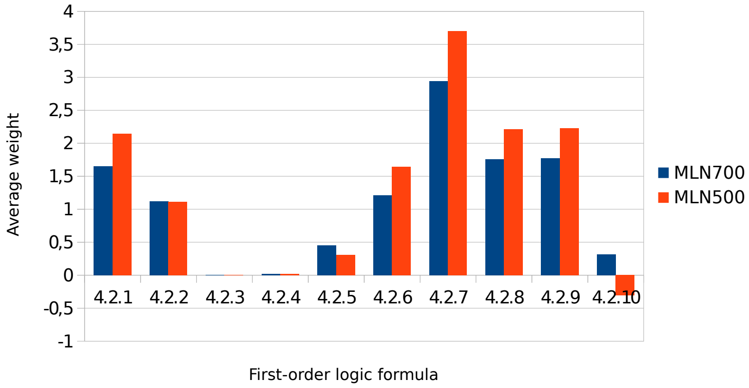


Figure 5.3.4: Average weights of the MLN500 and MLN700 models

different from the TRAIN500 sets. To observe the characteristics among the training sets, we compared the average weights of all the MLN500 and MLN700 models. Figure 5.3.4 highlights that the average weights of both MLN500 and MLN700 models are very much alike. In both MLN500 and MLN700 models, the continuation of the themes in the last utterance (Formula 4.2.1) is more likely than the prediction of a new theme in the next utterance (Formula 4.2.2). We used different strategies to predict an appearance of a new theme. From various training data sets, the models learned that a new theme is likely to appear after an occurrence of an Answer or an Acknowledgement in the last utterance (Formula 4.2.8 and 4.2.9). All the MLN models also tend to predict *Continuations* if there is a question in the previous utterance.

In the MLN500 models, Formula 4.2.10 obtains the lowest weight and it is rather unlikely to be true. In the MLN700 models, however, it is a bit more likely to be true. For both MLN500 and MLN700 models, Formula 4.2.3 and Formula 4.2.5 have a low tendency to be true, while Formula 4.2.7 has a the highest tendency compared to the other formulas. The average weights of some formulas including Formula 4.2.6, 4.2.7, 4.2.8 and 4.2.9 are smaller in the MLN700 models compared to in the MLN500 models. The fall in the weights seems to be caused by the greater variety of thematic changes in TRAIN700. The first-order logic formulas seem to be inconsistent with many interpretations of the additional ground atoms in the TRAIN700 sets.

Formulas with low or negative weights are still useful. We show in section 5.4.2 that the removal of Formula 4.2.3 and Formula 4.2.4 decreases the performance of MLN700 models.

5.4 Evaluation Results

In this section, we present the results of the evaluation of the MLN model's performance. First of all, in section 5.4.1, we exemplify successful and unsucessful MLN theme predictions. Then, in section 5.4.2, we describe and compare the performance of the MLN500 and MLN700 models over 5-fold cross validation using different dialogue history lengths. Moreover, we compare the model performance to random and informed baselines in section 5.4.3. In section 5.4.4, we also present the results of the MLN models in predicting each type of the annotated thematic changes. Finally, in section 5.4.5, we present an analysis of the errors produced by the MLN models in predicting theme.

5.4.1 MLN Theme Prediction

An MLN model is able to answer a question of what the theme in the next utterance will be. It takes a query and its dialogue context as its inputs, and outputs the probability of each possible theme in the next utterance. We describe the inputs and outputs of an MLN model in section 4.2.2 and 4.2.3. In evaluating an MLN model, we do not consider all the outputs of an MLN as its predictions. Instead, we only consider the MLN output with the highest probability and all other outputs whose probabilities are no more than 0.05 behind, as MLN predictions.

An MLN prediction of a thematic change can be of type *Continuations, Shift News* or *Shift Olds*. A predicted theme in the next utterance is correct, if it is identical with the annotation of that utterance. In this section, we illustrate two samples where an MLN model correctly predicts a *Continuation* and a *Shift New*. Additionally, we show an unsuccessful sample in predicting a *Shift Old*.

Dialogue history	Parker: Okay. We copy that. Schmitt: Do you need to take a vertical pan?
Next utterance	Cernan: Yeah, I've gotten it all.

Figure 5.4.1: The text dialogue of sample in Table 4.6

Figure 5.4.1 shows the text version of the sample query in Table 4.6. Schmitt, the speaker of the utterance *U669* asked about a pan, which means that he focused his attention to a pan. In the next utterance, Cernan gave an Answer which is also about the pan. The sample results in Table 4.7 are outputs of an MLN model trained with 700 utterances and given the query in Table 4.6. The MLN model predicts that there will be a *Continuation* in the next utterance. This prediction model is correct as the speaker of the next utterance indeed continued talking about the theme in the last utterance.

Dialogue history	Cernan: I'm on frame count 42. Parker: Copy, 42.
Next utterance	Cernan: Did you get a locator from here, Jack?

Figure 5.4.2: A sample of a *Shift New*

Figure 5.4.2 shows a sample dialogue text where the theme of the next utterance is a new theme. In the two utterances within the dialogue history, Cernan and Parker were talking about a frame count. The theme of the first utterance is continued in the second utterance as Parker gave an Acknowledgement about the frame count. In the next utterance, Cernan introduced a locator as a new theme because the locator was not a theme appearing in the previous utterances.

Possible Grounding	Probability
themeIn(U709,U707)	0.733
themeIn(U709,U709)	0.852

Table 5.4: Sample results of an MLN model for the sample in Figure 5.4.2

In our dialogue data, the first utterance of the sample in Figure 5.4.1, is annotated as the utterance *U707*. Given a query using the context from Figure 5.4.1, an MLN model predicts the theme of the utterance *U709*. Using the same MLN model as in the previous example, we show that the model is also able to predict an occurrence of a new theme correctly. The MLN model computes the probabilities of all possible themes of *U709*, which are shown in Table 5.4. Since the theme of *U708* is identical with the theme of *U707*, there are only two possible groundings of the predicate *themeIn* of *U709*. The MLN predicts with 85% probability that the theme in U709 will be new, which is correct. The probability of a new theme appearing in *U709* is higher than the probability of a continuation of theme *U707* in *U709*, because the MLN has learned that a new theme tends to appear after an occurrence of an ACKNOWLEDGEMENT.

Dialogue history	Cernan: Yeah, I'll get that. Schmitt: This fillet? Cernan: You got it? Parker: And, 17...
Next utterance	Schmitt: (To Houston) This is a fillet from underneath the rock.

Figure 5.4.3: A sample of *Shift Old*

Another type of thematic change is *Shift Old*. Figure 5.4.3 presents a sample text dialogue with dialogue history length four. In this sample, "fillet" is introduced as the theme of the second utterance in the dialogue history. It is continued as the theme of the third utterance. In the fourth utterance, which is the last utterance, the theme is shifted to "17", which is a new theme. The theme in the next utterance is again about "fillet". Therefore, there is a theme shift to the theme of the second and the third utterances.

Possible Grounding	Probability
themeIn(U727,U561)	0.468
themeIn(U727,U578)	0.472
themeIn(U727,U10)	0.695
themeIn(U727,U727)	0.610

Table 5.5: Sample results of an MLN model for the sample in Figure5.4.3

An MLN model computes the probabilities of all possible groundings of the theme in the next utterance. However, the probability of an old theme is typically lower than the probability of the theme of the last utterance or the probability of a new theme. Table 5.5 shows the results of an MLN model given the query using the sample in 5.4.3. The dialogue history starts from utterance *U723* whose theme is in *U561*. *U724* and *U725* have the same theme which

History Size	Measure	5-Fold Cross Validation					Mean	Std. Dev
		k1	k2	k3	k4	k5		
1	P	0.622	0.515	0.533	0.602	0.600	**0.574**	0.047
	R	0.623	0.508	0.543	0.602	0.602	0.576	0.048
	F1	0.622	0.511	0.538	0.602	0.601	0.575	0.048
2	P	0.609	0.515	0.524	0.557	0.591	0.559	0.041
	R	0.615	0.508	0.523	0.730	0.608	0.597	0.089
	F1	0.612	0.511	0.524	0.632	0.599	0.576	0.055
4	P	0.604	0.519	0.519	0.551	0.576	0.554	0.037
	R	0.612	0.518	0.523	0.743	0.657	**0.611**	0.095
	F1	0.608	0.518	0.521	0.633	0.614	**0.579**	0.055
6	P	0.594	0.519	0.510	0.536	0.567	0.545	0.035
	R	0.602	0.518	0.508	0.747	0.650	0.605	0.099
	F1	0.598	0.518	0.509	0.624	0.605	0.571	0.053
Average	P	**0.607**	0.517	0.522	0.562	0.584	0.558	0.040
	R	0.613	0.513	0.524	**0.706**	0.629	0.597	0.083
	F1	0.610	0.514	0.523	**0.622**	0.605	0.575	0.056

Table 5.6: The performance of the MLN 500 models over 5-fold cross validation

is in U578. The theme of *U726* is in *U10*. The annotation suggests that the theme in *U727* is *U578* (47%). However, the MLN yields higher probabilities for a continuation of the theme of *U726* (69%) and for a new theme in *U727* (61%). Thus, according to the annotation, the MLN model fails to predict the theme of *U727*.

We ran experiments with different MLN models. Some models are only able to predict theme continuations and appearances of new themes. Nevertheless, some other models are also able to predict a theme shift to an old theme. We discuss the performance of these MLN models in the next sections.

5.4.2 MLN Model Performance

Using the different TRAIN500 and TRAIN700 sets described in 5.2, we created 10 MLN500 models and 10 MLN700 models. We used different dialogue history sizes (i.e. one, two, four, and six) and tested all the MLN models against five test data sets. In total, we have run 40 experiments and computed precision, recall and F1 score (i.e. performance measures) for the results of each experiment. We summarize the precision, recall and F1 score values of the MLN500 models in Table 5.6 and those of MLN700 models in Table 5.7. For each dialogue history size, we computed average performance measures over the 5-fold cross validation. Moreover, we also computed the standard deviation of the values.

Over all dialogue history sizes, the MLN500 models yield the highest precision in the first round of cross validation $k1$. The MLN700 models, on the other hand, obtain the highest precision in $k5$. The highest recall and F1 score of the MLN500 models are reached in $k4$, while those of the MLN700 models are reached in $k1$. On average over all 5-fold cross validation, both MLN500 and MLN700 precision decreases as the dialogue history size increases. The average recall of the MLN500 models goes up from dialogue history lengths

History Size	Measure	5-Fold Cross Validation					Mean	Std. Dev
		k1	k2	k3	k4	k5		
1	P	0.620	0.600	0.532	0.583	0.608	**0.589**	0.034
	R	0.617	0.569	0.547	0.587	0.605	0.585	0.028
	F1	0.618	0.584	0.539	0.585	0.607	0.587	0.030
2	P	0.577	0.559	0.523	0.567	0.604	0.566	0.029
	R	0.665	0.694	0.563	0.615	0.618	0.631	0.050
	F1	0.618	0.619	0.543	0.590	0.611	0.596	0.032
4	P	0.567	0.545	0.520	0.537	0.592	0.552	0.028
	R	0.712	0.727	0.608	0.665	0.612	0.665	0.055
	F1	0.631	0.623	0.561	0.594	0.602	**0.602**	0.028
6	P	0.570	0.534	0.509	0.543	0.579	0.547	0.028
	R	0.720	0.708	0.603	0.690	0.607	**0.666**	0.056
	F1	0.636	0.609	0.552	0.608	0.593	0.600	0.031
Average	P	0.584	0.560	0.521	0.558	**0.596**	0.563	0.030
	R	**0.679**	0.675	0.580	0.639	0.611	0.637	0.047
	F1	**0.626**	0.609	0.548	0.594	0.603	0.596	0.030

Table 5.7: The performance of the MLN700 models over 5-fold cross validation

one to four, but then goes down for dialogue length six. In the case of the MLN700 models, the average recall keeps increasing as the dialogue history length increases. The F1 scores for both MLN500 and MLN700 models reach their peaks using a dialogue history length of four. Over all dialogue history lengths and the 5-fold cross validation, the average performance measures of the MLN700 models are higher than those of the MLN500 models. Moreover, the standard deviations of the performance measures of the MLN700 models are smaller than those of the MLN500 models. The standard deviations of both the MLN500 and MLN700 models are lower than 0.1. This suggests that the results of the MLN models over all experiments are pretty stable.

Over all experiments, the MLN700 models yield better theme predictions than the MLN500 models. The precisions of MLN700 models are similar to those of MLN500 models, but their precisions are slightly higher on average. Their average recall and F1 score values are 4% and 2% higher than those of the MLN500 models.

In section 5.3, we discussed that the weights of the first-order logic formulas in the MLN500 models trained using $k4$ training data sets are lower than using other training data sets. We also discussed that the weights themselves do not directly reflect model performance. Instead, the distribution of the weights should be considered. In Table 5.6, we show that the performance of the MLN500 models in $k4$ yield the highest average recall and F1 score. The average precision is also higher than the average precision over all k. The weights learned using the training data set in $k4$ are more equally distributed than in other rounds of cross validation.

Since the average performance of MLN500 models in $k4$ is better than the others, we suppose that the MLN500 models in other rounds suffer from overfitting problems. The problem might be because there is strong evidence for some rules and lack of evidence for other rules. The rules which characterize thematic changes for specific patterns, such as an occurrence of an Acknowl-

EDGEMENT, get lower weights in $k4$. In contrast, the more general rule characterizing a theme re-occurrence get a weight about 6 times higher in $k4$ than in most other rounds. These phenomena also appear in the cross validation for MLN700 models. In section 5.3, the three MLN700 models yielding the best F1 scores are shown to have similar phenomena regarding the much higher weight of the theme re-occurrence rule relative to the other rules.

Moreover, in section 5.3, we show that some first-order logic formulas of an MLN model are associated with low weights. For instance, Formula 4.2.3 and Formula 4.2.4 have low weights in the MLN700 models. To examine the impact of the first-order logic formulas that have low weights, we generated MLN models without Formula 4.2.3 and Formula 4.2.4 using the TRAIN700 sets. We call these models MLN700'. We tested them using our test sets with dialogue history length four. Then, we compared the performance of the MLN700' models to that of the MLN700 models containing Formula 4.2.3 and Formula 4.2.4. We computed the performance measures of the MLN700' models. Precision, recall and F1 score are all the same, that is 0.538. The precision (0.552), the recall (0.665) and the F1 score (0.602) of the MLN700 models are higher that those of the MLN700'.

5.4.3 Comparison to Baselines

We compare the performance of the MLN500 and MLN700 models to that of random baselines. A random baseline randomly chooses a theme from all existing themes in a given dialogue history or a new theme. The probability of a random baseline to correctly predict the themes in the next utterance depends on the dialogue history length. Since a longer dialogue history tends to contain more themes than in a shorter dialogue history, the random baseline performance goes down from dialogue history lengths one to six.

We also compare the MLN models to an informed baseline which always predict all the themes in the last utterance to be the theme of the next utterance. Unlike for the random baseline, the informed baseline performance is not affected by the different lengths of a dialogue history. This is because an informed baseline is only informed about the last utterance. Thus, it always produce the same predictions regardless of the length of a dialogue history.

For both random and informed baselines, we computed precision, recall and F1 score values for each test data set using different dialogue history lengths. For each dialogue history length, we compute the average values of the performance measures. Table 5.8 suggests that both the MLN500 and the MLN700 models outperform the random baseline. Averaged over all 5-fold cross validation datasets and all different history lengths, the MLN500 models yield approximately 17.4% higher precision (0.558) and 23% higher recall (0.597) than the random baseline which yields precision 0.385 and recall 0.368. Besides, the MLN700 models yield approximately 18% higher average precision and 27% higher average recall values than the random baseline. According to Wilcoxon signed rank tests, the precisions and recalls of the MLN500 and MLN700 models are significantly higher than those of the random baselines generally at p-value < 0.01.

Both the MLN500 and the MLN700 models also yield slightly better results than the informed baseline. The MLN500 models yield 2% higher average precision and 6% higher average recall compared to the precision (0.538) and the

History Size	Average Performance	MLN500 Models	MLN700 Models	Informed Baseline	Random Baseline
1	P	**0.574**	**0.589**	0.538	0.510
	R	0.575	0.585	0.538	0.489
	F1	0.575	0.587	0.538	0.500
2	P	0.559	0.566	0.538	0.417
	R	0.597	0.631	0.538	0.400
	F1	0.576	0.596	0.538	0.408
4	P	0.554	0.552	0.538	0.352
	R	**0.611**	0.665	0.538	0.333
	F1	**0.579**	**0.602**	0.538	0.342
6	P	0.545	0.547	0.538	0.259
	R	0.605	**0.666**	0.538	0.248
	F1	0.571	0.600	0.538	0.253
Average	P	0.558	**0.564**	0.538	0.385
	R	0.597	**0.637**	0.538	0.368
	F1	0.575	**0.596**	0.538	0.376

Table 5.8: Average performance of MLN500 and MLN700 models compared to the Informed and the Random baselines

recall (0.538) of the informed baseline. Besides, the MLN700 models yield approximately 3% higher average precision and 10% higher average recall values. The average F1 score (0.575) of the MLN500 models are about 4% higher than that of the informed baseline and 20% higher than that of the random baseline. The F1 score of the MLN700 models is 22% better than that of the random baseline and 6% better than that of the informed baseline.

The precisions and recalls of the MLN700 models are generally significantly higher than those of the informed baseline at p-value < 0.20. Moreover, precisions of the models using the dialogue history size 1 are usually more significant than using other sizes. In contrast, the recalls are more significant using a larger size. The precisions and recalls of the MLN500 models using the dialogue history size 1 are also significantly higher than those of the informed baseline at p-value < 0.20.

We suspect that the performance of MLN models are not much better than the informed baseline, because the training data is still insufficient for learning proper weights of the first-order formulas. Using the TRAIN700 data sets does not improve the results much in comparison to the use of the TRAIN500 data sets. The additional 200 utterances in a TRAIN700 data set seem to be insufficient to considerably improve representing the variety of thematic changes. Besides, the MLN models may need more clues to be able to predict an occurrence of a new theme in the next utterance and to be specific in its predictions. The MLN models might be able to predict a thematic change correctly but at the same time it may produce errors by over-predicting another theme also. We discuss more about the errors produced by the MLN models in section 5.4.5.

History Size	Theme Prediction	Annota-tion	MLN500	MLN700
1	*Continuation*	168.40	140.80 (83.61%)	132.20 (78.50%)
	Shift New	144.00	38.40 (26.67%)	49.20 (34.17%)
	Shift Old	0.00	0.00 (0.00%)	0.00 (0.00%)
2	*Continuation*	168.40	142.40 (84.56%)	137.20 (81.47%)
	Shift New	128.60	43.80 (34.06%)	**59.80 (46.50%)**
	Shift Old	17.40	0.00 (0.00%)	0.00 (0.00%)
4	*Continuation*	168.40	150.20 (89.19%)	151.40 (89.90%)
	Shift New	115.00	40.40 (35.13%)	56.60 (49.22%)
	Shift Old	32.00	0.00 (0.00%)	0.00 (0.00%)
6	*Continuation*	168.40	151.20 (89.79%)	**155.20 (92.16%)**
	Shift New	108.60	37.60 (34.62%)	53.00 (48.80%)
	Shift Old	39.80	0.40 (1.01%)	**0.60 (1.51%)**
Average	*Continuation*	168.40	**146.15 (86.79%)**	144.00 (85.51%)
	Shift New	124.05	40.05 (32.62%)	**54.65 (44.67%)**
	Shift Old	22.30	0.10 (0.25%)	**0.15 (0.38%)**

Table 5.9: Correct predictions MLN500 and MLN700 models identical to the annotated themes by thematic change type

5.4.4 Predicting Thematic Change Type

Precision gives us information about how precise or accurate the predictions of an MLN model are, and recall demonstrates the completeness of the predictions with respect to covering all the themes in the annotation data. Both precision and recall statistically measure the performance of an MLN model, but they do not specifically measure the performance in modeling thematic change types. We measure how well an MLN model predicts each type of thematic change, by computing the recall of each thematic type. Note that this measurement ignores the errors an MLN model might produce, and only takes into account the correct predictions of the MLN.

First of all, for each test data set with each dialogue history size, we computed the number of each thematic change type in the annotation. Likewise, we also computed the number of each thematic change type of the correct predictions of the MLN500 models and those of the MLN700 models. Then, we calculated the average number of each thematic change for each dialogue history size. In Table 5.9, we present the average number of each thematic change in the annotation and in the correct predictions of the MLN500 and the MLN700 models. We also show the percentage of correct predictions for each thematic change type, that is, the recall of each thematic change type.

Since an utterance may contain more than one theme, the average total number of all annotated themes (312.4) in all test data sets is larger than the total number of utterances in the test data sets (300). On average, all the test data sets with different dialogue history lengths contain 54% *Continuations*, 40% *Shift News*, and rather rarely, 7% *Shift Olds*. There are 168.4 *Continuations* regardless of the different history length. Moreover, there are 124.05 *Shift News* and 22.30 *Shift Olds* on average.

The number of *Shift News* and *Shift Old*s are relative to the length of the dialogue history. A dialogue history of size 1 only contains the last utterance, so a theme in the next utterance can only be a *Continuation* or a *Shift New*, but not a *Shift Old*. As the dialogue length increases, a *Shift Old* may be a theme in the next utterance and its probability increases. A *Shift Old* relative to a longer dialogue history always replaces a *Shift New* relative to a shorter dialogue history. This is because it would be a *Shift New* if only a smaller dialogue history length is considered. The number of *Shift Olds* in a test data set will become greater given a longer dialogue history, but the number of *Shift New*s will become smaller.

Over all dialogue history lengths, the MLN models correctly predict up to nearly 87% of all *Continuations* on average. The average successful *Continuation* prediction of the MLN500 models (86.79%) is slightly better than that of the MLN700 models (85.51%). The reason for this is discussed in section 5.4.5. The MLN700 models (44.67%), however, predict more *Shift New* occurrences than MLN500 models (32.62%).

The MLN models are able to correctly predict more *Continuations* when larger dialogue history lengths are considered. As the dialogue history size is increased, the number of correct *Continuation* predictions grows. Therefore, the MLN models obtain the best predictions of *Continuations* using a dialogue history length of six utterances. The number of correct MLN model predictions of *Shift New* jump up when using a dialogue history length of two instead of one, but then fall again, when using dialogue history lengths between two and six. Thus, the MLN models obtain their largest amount of identified *Shift New* predictions by using a dialogue history length of two utterances.

Since the informed baseline always continues a theme of the last utterance to be a theme of the next utterance, it is able to predict 100% *Continuation*, but 0% *Shift New* and *Shift Old*. The MLN models correctly predict less *Continuations* than the informed baseline, because the MLN models consider the probability of other thematic change types. The models are able to predict up to 86.79% of the *Continuations* and almost 45% of the *Shift New* occurrences. Moreover, the MLN models are shown to be able to find a *Shift Old*, although, in the best result, using a dialogue history length of six utterances, only 0.38% of all *Shift Old* occurrences are correctly identified. This is interesting, since the first-order logic formulas do not include a particular rule for predicting a *Shift Old*.

5.4.5 Error Analysis

A typical error of an MLN model is to predict a *Shift New*, when there is a *Continuation* in the annotation. This type of error occurs because the MLN models have a fairly high tendency to predict a new theme after an occurrence of an ANSWER or an ACKNOWLEDGEMENT. Although the prediction may be true in many cases, it is not true in all cases. On the other hand, the MLN models might also produce a wrong prediction of a *Continuation* while there is a *Shift New* in the annotation. Predicting a new theme is difficult when there is no explicit clue for the MLN models. For instance, there is no ANSWER or ACKNOWLEDGEMENT in the last utterance.

Although the MLN models succeed in predicting most of the *Continuations* and almost half of the *Shift New*s, the precisions are not as high as one might expect. This is because the MLN models might have over-predicted the theme of the next utterance. In other words, the models might produce more predictions than they should. In this case, the MLN models fail to predict only a partial continuation of the themes in the last utterance, when not all themes are continued to be talked about. Moreover, while predicting a *Shift New*, the MLN models are also likely to predict a *Continuation*. When there is actually only a *Shift New* in the annotation, the MLN models might produce too many predictions.

On the contrary, it may also happen that the MLN models make too few predictions which results in low recalls. The MLN models tend to predict *Continuation*s only, whereas a *Shift New* may occur together with a *Continuation* in the annotation. Therefore, sometimes the models do not predict enough themes in the next utterance.

Predicting a *Shift Old* is a difficult task since it seems that there is no overt clues that can be used, at least with the way the data is annotated in this work. The MLN models do not contain specific formulas to predict a *Shift Old*. Even so, the models still consider the recent themes and compute their probabilities to appear in the next utterance. The MLN models mostly predict a *Continuation* or a *Shift New*. However, the models may predict a *Shift Old*, when a theme has been continued a few times in a given dialogue history. The MLN models need a longer dialogue history to believe that the theme that has been talked about continuously in the past, but does not occur in the last utterance, is likely to be talked about in the next utterance. We show in 5.9, the MLN models can predict a *Shift Old* by using the dialogue history size six.

Both the MLN500 and the MLN700 models produce similar errors to the ones that have been discussed so far. Nevertheless, the MLN700 models ameliorate the errors better than the MLN500 models. This is because the MLN700 models tend to produce more *Shift New* predictions than the MLN500 models. While predicting a *Continuation*, the MLN700 models also predict a *Shift New*. In this way, the models are able to capture more *Shift News* in the annotation than MLN500 models (see Table 5.9). Unfortunately, this also results in more failures in predicting a *Continuation* compared to the MLN500 models. When the theme of the next utterance is supposed to be a *Continuation*, and the MLN500 models predict it correctly, the MLN700 models predict a *Shift New* instead.

Chapter 6

Discussions

In section 6.1, we describe the solutions given by an MLN model to solve the problem of modeling thematic changes. Moreover, we also present the difficulties of using an MLN model. In section 6.2, we show how an MLN model can be used in the setting of human-robot interaction in performing a USAR task. Finally, in section 6.3, we explain how an MLN model can be applied to improve various modules in a dialogue system.

6.1 Solutions and Difficulties

MLN models offer a solution to structuring thematic changes in a dynamic dialogue, where the dialogue themes change throughout the dialogue without following a certain order or pattern. MLN models can do this because they consider all possible themes in a given dialogue history, and a new theme, as possible next themes. Therefore, the thematic structure represented by an MLN is more flexible than a stack model and the centering theory (see section 2.3.2 and section 2.3.4). Unlike a stack model and the centering theory, MLN models do not restrict possible connections/relations between the themes of different utterances. In MLNs, the next theme does not only depend on the last utterance, but on all utterances in a given dialogue history. Each theme of these utterances may be related to the themes in the next utterance.

Like a knowledge-based approach, MLN models can explain the characteristics of thematic changes (i.e. the decision-making process behind the activation of a thematic change). By using logical rules (i.e. first order-logic formulas), MLN models describe how a thematic change type may occur. For example, a *Continuation* may occur after a question. While a knowledge-based approach uses an ontology to guide how a dialogue is likely to progress (see section 2.3.6), MLN models do not restrict a dialogue to following a certain pattern. The logical rules of MLN models are not applied in a certain order or pattern. Thus, MLN models would not suffer from pattern complexity problems when dealing with a complex dialogue.

Moreover, MLN models also use statistics to assign weights to the logical rules. This allows MLN models to learn the tendencies of thematic change characteristics from data. Thus, MLN models do not only assume the characteristics of thematic changes, but reflect how the characteristics actually occur in reality. Like a typical statistical model, however, MLN model performance is highly influenced by the training data that was used. Since thematic changes in different training data sets can have different characteristics, MLN models may reflect different tendencies of the characteristics of the training data. To obtain representative estimations of the tendencies, MLNs need a large amount of training data containing a lot of evidence of thematic change characteristics. MLNs are vulnerable to over-fitting problems due to insufficient training data. A small training data set may result in biased weights of the formulas in MLN models, due to irregular occurrences of thematic changes.

Not only can MLN models describe the thematic changes in a dialogue, they can also predict the next dialogue theme at some point in a dialogue. MLN models yield theme predictions by computing the probability of each possible dialogue theme in a given dialogue history being the next theme. Additionally, the models also compute the probability of a new theme in the next utterance. MLN models do not strictly predict a single theme, ignoring any possible additional themes. This way of predicting theme is more realistic, because an utterance may express more than a single dialogue theme.

MLN models consider all three types of thematic changes, namely *Continuation*, *Shift New* and *Shift Old*, as its predictions. However, they rarely correctly predict a *Shift Old*. Theoretically, the characteristics of a *Shift Old* can be formulated as a first-order logic rule. However, the occurrence of a *Shift Old* itself is not regular and does not really show any pattern. Thus, it is hard to discover *Shift Old* characteristics and define a rule to predict an occurrence of a *Shift Old*. We suspect that annotation is probably insufficient to allow us to come up with such a rule. Since estimating the probability of a *Shift Old* is difficult without a rule, predicting its occurrence correctly is also difficult.

A difficulty in predicting the next theme is to predict the number of themes. Since an utterance may express more than a single dialogue theme, an MLN model should be able to predict more than one theme. However, it is rather difficult to approximate how many dialogue themes will appear in the next utterance. Sometimes an MLN estimates too many themes with high probabilities. In other cases, an MLN yields rather low probabilities for all possible themes and only one theme gets higher probabilities than the others. In practice, the use of a system determines how precise the number of dialogue themes should be modeled. Predicting at least one correct dialogue theme can be adequate in some systems to keep a conversation going. On the other hand, some systems may require a more precise number of predictions to be able to recognize the themes in the next utterance and properly understand the intention of a speaker.

Dialogue data typically contains noise. MLN models are useful for describing noise characteristics and creating strategies for handling noise. In this work, we represent noise as a dialogue theme of type OTHER. In fact, the thematic changes after an occurrence of an OTHER theme are varied and not regular. Therefore, it is rather hard to be characterized. Moreover, the noise characteristics in different data can be different. For instance, a new theme may be likely to appear after noise in some data sets, but it may not be the case in other data sets. This should be taken into account to formulate logical rules for handling noise.

6.2 Thematic Structure for Human-Robot Interaction in USAR

In this work, we attempt to build a thematic structure which can describe and predict thematic changes in human-robot interaction, especially in performing a USAR task. Due to the lack of human-robot dialogue data, however, we developed our approach based on a human-human dialogue data set. We carefully chose a human-human dialogue data set (i.e. the Apollo 17 transcript) that is akin to a human-robot dialogue data set. The similarities between the data set we use and a human-robot dialogue data set have been described in section 4.1. Since the characteristics of thematic changes in the Apollo 17 transcript are considered similar to those in a human-robot data set, we expect that our approach can also be used for structuring the dialogue themes in human-robot data sets. Besides, the approach has been shown to model a complicated human-human dialogue. Since a human-robot dialogue is simpler than a human-human conversation, the approach can be applied to model a human-robot dialogue as well.

A human-robot dialogue is simpler than a human-human conversation, because a robot has much more limited capabilities than a human. A human-robot dialogue usually use a controlled language is typically simpler than a natural language in general. The grammar and vocabulary in a human-robot dialogue is reduced to avoid ambiguity and complexity. Moreover, a human-robot dialogue does not contain a lot of problem solving, such as how to perform a task or deal with a problem/obstacle while performing a task. Instead, it typically contains a lot of instructions from a human operator and reports from a robot. Human behavior, when talking with a robot, tends to be different from when talking with a human. For example, people often compliment and encourage each other, but a human operator may not bother to do so to a robot. Besides, people may ramble or joke in a human-human conversation, but it is not likely to happen in human-robot interaction. A human operator should be aware of the limitation of a robot, and therefore he will be likely to use simple commands to talk to a robot.

Furthermore, a USAR task is typically performed in a stressful situation, where a rescuer must perform a task in a dangerous place with limited time. A conversation in such a situation must not be verbose, but short and concise. The Apollo 17 mission is actually also constrained with time. The tasks in the mission were carefully planned with time slots, because the astronauts have limited oxygen and other resources. Despite the time limit, however, the dialogue participants still sometimes produced an utterance which is irrelevant to performing a task. In a task-oriented dialogue, such an utterance is considered as noise, because it drifts a conversation away from focusing on a task. In this work, we do not clean up the data by removing such noise, but treat it as a theme of type OTHER. By doing so, we keep the rich dialogue data, but simplify its representation.

Human-robot data does not tend to contain such irrelevant talking, but it may contain noise due to errors produced in some modules of the robot. For instance, an utterance of a human operator is not clear because of background noise. As a result, the speech recognition module cannot recognize it. In fact, the Apollo 17 data also contains incomplete transcriptions for such a case. This reflects that a human may also be unable to recognize such an utterance. The way we treat noise in annotating the Apollo 17 data, can be applied to treating noise in human-robot data. In other words, noise in human-robot interaction can be assigned as a theme of type OTHER.

As mentioned in section 1.4, a robotic system deals with a lot of uncertainty in all of its modules. A module in a robotic system usually tries to approximate the best possible results as its outputs. For instance, an image recognition module tries to recognize an object, but it may not be certain what the object is. Therefore, the results of the module are an approximation of the real object. Sometimes, the target to approximate is uncertain. For instance, what a robot should say in response to a user utterance is uncertain. A robot may give different responses to a user utterance. Some of these responses may be preferable to others, but the other responses may still be acceptable. Similarly, more than one theme could be the next theme of a conversation. MLN models can address this issue, as they estimates the probability (i.e. the degree of certainty) of all the recent themes in a dialogue history, and that of a new theme being the next dialogue theme.

6.3 Application

In this work, we use annotated data for illustrating and evaluating MLN models. To use an MLN model in a dialogue system or a robotic system, theme identification and speech act classification modules are needed. A theme identification module identifies the TASK, ENTITY and/or OTHER contained in an utterance. On the other hand, a speech act classification module assigns a speech act to an utterance, if it is a QUESTION, an ANSWER or an ACKNOWLEDGMENT. Moreover, PAUSES in a dialogue should also be identified.

Since an MLN model takes inputs in first-order logic form, one also needs a first-order logic conversion module to transform an annotated text to grounded atomic formulas. As described in section 4.2.2, the negative atomic formulas for all the remaining possible groundings should also be explicitly specified. The first-order conversion module should also generate these negative atomic formulas. Given a database of ground atoms, an MLN tool such as Alchemy can be used to learn weights for the first-order logic formulas described in section 4.2.1. Moreover, it can also be used to run inference using weighted first-order logic formulas (i.e. to produce MLN predictions).

An MLN-based thematic structure which can predict the next dialogue themes can be used in various ways. As mentioned in 1.1, such a thematic structure can help ASR to reduce speech recognition errors. Thus, it will help a dialogue system to better understand a user utterance. Predicted themes can be used to guess the keywords in a user utterance, when a user utterance is noisy and ASR cannot recognize the words in the user utterance well. Besides, they can provide a context for priming ASR outputs. For such a purpose, the predicted themes are associated with a bag of words related to the themes. These words are then used to recognize or to expect the words in the next utterance of the user. Moreover, they can be used to prime unexpected ASR outputs.

Predicted themes can also be used to predict the intention conceived in an utterance. For instance, a TASK describes the intention of a speaker about performing the TASK. Moreover, we use a prediction of a *Shift New* to expect a move in a plan or a task list. Typically, a USAR team has a check list of the tasks they have to do. When an MLN model predicts a *Shift New*, we can use the task list to further predict what the new theme could be.

A robot discourse planner can benefit from an MLN model to produce a system response related to a predicted theme. When the natural language module of a robot fails to understand a user utterance, an MLN model can be used to guide a clarification sub-dialogue, e.g. asking for a repair of a user utterance. A clarification sub-dialogue has the purpose of clarifying whether the user has talked about the predicted theme with the highest probability. If the robot prediction is correct, the sub-dialogue further progresses on the theme. In this way, a robot shows some degree of expectation or idea about a user utterance, and resolve a non/partial-understanding of the utterance.

Chapter 7

Conclusions

This chapter presents a conclusion of our work and a discussion of future work.

7.1 Conclusions

We have described an approach to modeling the changes of dialogue themes (i.e. thematic changes) in a collaborated task-oriented dialogue. We have introduced a categorization of a dialogue theme into three categories, namely ENTITY, TASK, and OTHER. This categorization allows us to describe the intention of a speaker and use it to structure the thematic changes in a conversation. Moreover, noise can be represented and handled by using the OTHER type.

We have introduced three types of thematic changes, which are *Continuation*, *Shift New* and *Shift Old*. These thematic change types provide information that can help various modules in a dialogue system. We use an MLN to model the characteristics of thematic changes in a dialogue. This includes how a dialogue theme changes with respect to an occurrence of some speech acts, namely QUESTION, ANSWER, and ACKNOWLEDGMENT. Moreover, we have also specified how a dialogue theme may change with an occurrence of a PAUSE.

Our experiment results show that the MLN models are able to predict the thematic changes in the next user utterance. We show that, on average, the MLN models are able to predict up to 87% of all *Continuations* and 45% of all *Shift News* in the annotation. However, the models are not good in predicting a *Shift Old* and are only able to predict a *Shift Old* by using a dialogue history length of six utterances. Furthermore, we have shown that the MLN models outperformed both a random and an informed baseline. Although we used a human-human dialogue data set for developing and evaluating MLN models, we have argued that the approach can also be used for human robot interaction dialogue data.

7.2 Future work

The approach described in this thesis has been shown to be able to model thematic changes in a collaborative task-oriented dialogue. The MLN models, however, still suffer from several drawbacks, such as difficulties in predicting *Shift Old* and in handling noise. MLN model performance can be improved by adding more rules and training data. In crafting rules, however, one should keep simplicity, because complex rules will cause exponential complexity in learning weights. To characterize the occurrences of *Shift Olds* and to handle noise, the training data might also need a richer annotation. As overfitting problem is typical in statistical models, more training data is desirable to obtain representative weights for the logical rules of the MLN models.

As mentioned in section 1.3, a USAR task usually comprises well-structured sub-tasks. The logical rules in an MLN can be extended by including a task structure (i.e. an ontology of how a task is done step by step). A task structure can be represented as logical rules and used to support the predictions of the next theme. In this work, the MLN models can predict an occurrence of a new theme, but they do not predict what the new theme could be. This is because the MLN models are only informed by a dialogue history. A task structure can provide hints about possible ENTITIES and TASKS in an utterance which has not been perceived so far. Thus, this information could be used to specify a new theme and also to help predicting the next theme in general.

Moreover, MLN models can also benefit from a topological abstraction of the setting of a situated dialogue. Dialogue themes in a situated dialogue are influenced by the environment of the dialogue participants. Therefore, a topological abstraction can be useful to support the predictions of the next theme. A topological abstraction can provide information about possible ENTITIES and TASKS in a certain location. Together with a task structure, it can be used to predict ENTITIES and TASKS which might be talked about at a certain location.

Bibliography

J. L. Austin. *How to do things with words.* Harvard University Press, Cambridge, Mass., 1975.

Stephan Busemann, Stephan Oepen, Elizabeth Hinkelman, Günter Neumann, and Hans Uszkoreit. Cosma - multi-participant nl interaction for appointment scheduling. Research report, DFKI, Saarbrücken, 1994.

Christian Chiarcos, Berry Claus, and Michael Grabski. Introduction: Salience in linguistics and beyond. In *Salience: Multidisciplinary Perspectives on its Function in Discourse*, pages 1–28, Berlin, 2011. Walter de Gruyter GmbH.

Herbert H. Clark. *Using Language.* Cambridge University Press, 1996.

Herbert H. Clark and Edward F. Schaefer. Contributing to discourse. *Cognitive Science*, 13(2):259–294, 1989.

L.T. F. Gamut. *Introduction to Logic*, volume 1 of Logic, Language and Meaning. University of Chicago Press, Chicago, 1991.

H. P. Grice. Meaning. *The Philosophical Review*, 66:377–388, July 1957.

Barbara J. Grosz. Focusing and description in natural language dialogues. In I. Sag A. K. Joshi and B. Webber, editors, *Elements of Discourse Understanding.* Cambridge University Press, Cambridge, England, 1981.

Barbara J. Grosz and Candace L. Sidner. Attention, intentions, and the structure of discourse. *Computational Linguistics*, 12:175–204, July 1986.

Barbara J. Grosz, Scott Weinstein, and Aravind K. Joshi. Centering: A framework for modeling the local coherence of discourse. *Computational Linguistics*, 21:203–225, 1995.

M. A. K. Halliday and Christian M. I. M. Matthiessen. *An introduction to functional grammar.* Hodder Arnold, London, 3rd edition, 2004.

Graeme Hirst. *Anaphora in Natural Language Understanding: A Survey*, volume 119 of *Lecture Notes in Computer Science.* Springer, 1981.

E. M. Jones. The apollo 17 lunar surface journal. Technical report, USA, 1995.

Bernard J. Klaene and Russell E. Sanders. *Structural Firefighting: Strategiy and Tactics.* Jones and Bartlett Publishers, MA, USA, 2nd edition, 2008.

Stanley Kok, Marc Sumner, Matthew Richardson, Parag Singla, Hoifung Poon, Daniel Lowd, Jue Wang, and Pedro Domingos. The alchemy system for statistical relational AI. Technical report, Department of Computer Science and Engineering, University of Washington, Seattle, WA., 2009.

Alex Lascarides and Nicholas Asher. Segmented discourse representation theory: Dynamic semantics with discourse structure. *Computing Meaning*, 2007.

Hector J. Levesque, Philip R. Cohen, and José H. T. Nunes. On acting together. In *Proceedings of the eighth National conference on Artificial intelligence - Volume 1*, AAAI'90, pages 94–99. AAAI Press, 1990.

Pierre Lison. Robust processing of situated spoken dialogue. Master's thesis, Universität des Saarlandes, Saarbrücken, December 2008.

Pierre Lison, Carsten Ehrler, and Geert-Jan M. Kruijff. Belief modelling for situation awareness in human-robot interaction. In *Proceedings of the 19th International Symposium on Robot and Human Interactive Communication (RO-MAN 2010)*, 2010.

Robin R. Murphy, Satoshi Tadokoro, Daniele Nardi, Adam Jacoff, Paolo Fiorini, Howie Choset, and Aydan M. Erkmen. Search and rescue robotics. In *Springer Handbook of Robotics*, pages 1151–1173. Springer Verlag, 2008.

Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1988. ISBN 0-934613-73-7.

Ellen F. Prince. The zpg letter: Subjects, definiteness, and information-status. In S. Thompson and W. Mann, editors, *Discourse description: diverse analyses of a fund raising text.*, pages 295–325. John Benjamins, 1992.

Matthew Richardson and Pedro Domingos. Markov logic networks. *Mach. Learn.*, 62:107–136, February 2006. ISSN 0885-6125.

John Searle. *A taxonomy of illocutionary acts*, pages 334–369. University of Minnesota Press, Minneapolis, 1975.

John R. Searle. *Speech Acts: An Essay in the Philosophy of Language.* Cambridge University Press, 1969.

Candace L Sidner. Towards a computational theory of definite anaphora comprehension in english discourse. Technical report, Cambridge, MA, USA, 1979.

Robert Stalnaker. Common ground. *Linguistics and Philosophy*, 25:701–721, 2002.

Michael Strube. Never look back: An alternative to centering. In *COLING-ACL'98*, pages 1251–1257, 1998.

Michael Strube and Udo Hahn. Functional centering – grounding referential coherence in information structure. *Computational Linguistics*, 25:309–344, 1999.

David R. Traum and Elizabeth A. Hinkelman. Conversation acts in task-oriented spoken dialogue. *Computational Intelligence*, 8:575–599, 1992.

Wolfgang Wahlster, editor. *Verbmobil: Foundations of Speech-to-Speech Translation.* Artificial Intelligence. Springer, 2000.

Dietmar Zaefferer. On the coding of sentential modality. In Johannes Bechert, editor, *Toward a typology of European languages*, pages 215–237, Berlin; New York, 1990. Mouton de Gruyter.

Hendrik Zender, Geert-Jan M. Kruijff, and Ivana Kruijff-Korbayová. Situated resolution and generation of spatial referring expressions for robotic assistants. In *Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI-09)*, pages 1604–1609, Pasadena, CA, USA, July 2009.