Intrinsic and Extrinsic Approaches to Recognizing Textual Entailment

Dissertation
zur Erlangung des akademischen Grades eines
Doktors der Philosophie der Philosophischen
Fakultäten
der Universität des Saarlandes
vorgelegt von

Rui Wang

March, 2011

Abstract

Recognizing Textual Entailment (RTE) is to detect an important relation between two texts, namely whether one text can be inferred from the other. For natural language processing, especially for natural language understanding, this is a useful and challenging task. We start with an introduction of the notion of *textual entailment*, and then define the scope of the recognition task.

We summarize previous work and point out two important issues involved, meaning representation and relation recognition. For the former, a general representation based on dependency relations between words or tokens is used to approximate the meaning of the text. For the latter, two categories of approaches, intrinsic and extrinsic ones, are proposed. The two parts of the thesis are dedicated to these two classes of approaches. Intrinsically, we develop specialized modules to deal with different types of entailment; and extrinsically, we explore the connection between RTE and other semantic relations between texts.

In the first part, an extensible architecture is presented to incorporate different specialized modules handling different types of entailment. We start with one specialized module for handling text pairs with temporal expressions. A separate time anchoring component is developed to recognize and normalize the temporal expressions contained in the texts. Then it is shown that the generalization of this module can handle texts containing other types of named-entities as well. The evaluation results confirm that precision-oriented specialized modules are required.

We also describe another module based on an external knowledge resource. A collection of textual inference rules is applied to the RTE task after being extended and refined with a hand-crafted lexical resource. The evaluation results demonstrate that this is a precision-oriented approach, which can also be viewed as a specialized module. As alternative resources, we also present a pilot study on acquiring paraphrased fragment pairs in an unsupervised manner.

In the second part of the dissertation, a general framework is proposed to view textual entailment as one of the generalized *Textual Semantic Relations* (TSRs). Instead of tackling the RTE task in a standalone manner, we look at its connection to other semantic relations between two texts, e.g., paraphrase, contradiction, etc. The motivation of such a generalization is given as well as the framework of recognizing all these

relations simultaneously.

The prerequisites of the TSR recognition task are data and knowledge resources. An overview of all the corpora used for the experiments is given and followed by a discussion of the methodologies used in their construction. Then we elaborate on two corpora we constructed: one has a new annotation scheme of six categories of textual semantic relations with manual annotations; and the other uses a crowd-sourcing technique to collect the data from the Web.

After that, textual relatedness recognition is introduced. Although *relatedness* is usually user- and situation-dependent, in practice, it can help with filtering out the noisy cases. It is linguistically-indicated and can be viewed as a weaker concept than semantic similarity. In the experiments, we show that an alignment model based on the predicate-argument structures using relatedness as a measurement can help an RTE system to recognize the UNKNOWN cases (i.e. neither ENTAILMENT nor CONTRADICTION) at the first stage, and improve the overall performance in the three-way RTE task.

Finally the TSR classification is presented. A generalization of all the meaning representations described in the previous approaches is given. Then, a multi-dimensional classification approach is introduced, including *relatedness* as one of the dimensions. The other two are *inconsistency* and *inequality*. The approach is evaluated on various corpora and it is shown to be a generalized approach to entailment recognition, paraphrase identification, and other TSR recognition tasks. The system achieves the state-of-the-art performance for all these tasks.

As for the future work, we discuss several possible extensions of the current approaches. Some of the modules contained in the system have been already successfully applied to other natural language processing tasks. The promising results confirm the direction of research on this task and broaden the application area.

Zusammenfassung

Die Erkennung von texuellem Entailment (*Recognizing Textual Entailment*, RTE) ist das Aufdecken einer wichtigen Beziehung zwischen zwei Texten, nämlich, ob man den einen aus dem anderen schließen kann. RTE ist eine nützliche und herausfordernde Aufgabe für die automatische Verarbeitung natürlicher Sprachen im Allgemeinen und das maschinelle Sprachverstehen im Besonderen. Die Arbeit beginnt mit der Begriffserklärung und einer Definition der Erkennungsaufgabe.

Wir fassen bisherige Forschungsergebnisse zusammen und stellen dabei zwei wesentliche Themen heraus: Bedeutungsrepräsentation und Erkennung von Relationen. Für erstere benutzen wir eine allgemeine Repräsentation, die auf Dependanzrelationen zwischen Wörtern oder Token basiert, um die Bedeutung des Textes zu approximieren. Für die Relationserkennung werden zwei verschiedene Arten von Ansätzen vorgeschlagen: intrinsische und extrinsische. Die Dissertation gliedert sich in zwei Teile entlang dieser Unterscheidung. Im Rahmen der intrinsischen Ansätze entwicklen wir spezialisierte Module um verschiedene Arten von Entailment zu behandeln, mit den extrinsischen Ansätzen untersuchen wir die Verbindung von RTE und anderen semantischen Relationen zwischen zwei Texten.

Der erste Teil präsentiert eine erweiterbare Architektur, die unterschiedliche spezialisierte Module für unterschiedliche Arten von Entailment integriert. Wir beginnen mit einem spezialisierten Modul, welches Text-Paare mit temporalen Ausdrücken behandelt. Für die Erkennung und Normalisierung von temporalen Ausdrücken wurde eine separate Zeitverankerung-Komponente entwickelt. Dann zeigen wir, dass eine Verallgemeinerung dieses spezialisierten Moduls auch Texte mit anderen Arten von Eigennamen verarbeiten kann. Die Evaluationsexperimente zeigen, dass präzisionsorientierte spezialisierte Module erforderlich sind.

Wir stellen weiterhin ein Modul vor, welches auf einer externen Wissensressource basiert. Eine Reihe von Folgerungs-Regeln wird mit Hilfe einer manuell erstellten lexikalischen Ressource erweitert und verfeinert, um dann auf die RTE-Aufgabe angewendet zu werden. Die Evaluationsexperimente verdeutlichen, dass es sich dabei um einen präzisionsorientierten Ansatz handelt, welcher auch als ein spezialisiertes Modul betrachtet werden kann. Als alternative Ressourcen prsentieren wir eine Pilotstudie, in der wir paraphrasierte Fragment-Paare in einem unüberwachten

Ansatz gewinnen.

Der zweite Teil der Dissertation präsentiert ein allgemeines Rahmenwerk, in dem textuelles Entailment als Sonderfall von textuellen semantischen Relationen (*Textual Semantic Relation*, TSR) betrachtet wird. Statt das RTE-Problem isoliert zu bearbeiten, betrachten wir die Gemeinsamkeiten mit anderen semantischen Relationen zwischen zwei Texten, zum Beispiel Paraphrase, Kontradiktion, usw. Wir erläutern die Motive für eine solche Verallgemeinerung und präsentieren ein Rahmenwerk, um alle solchen Relationen simultan zu erkennen.

Die Voraussetzung für die TSR-Erkennung sind Daten- und Wissensressourcen. Wir geben einen Überblick über alle Korpora, die wir für die Experimente benutzt haben und diskutieren die Methoden zur Erstellung solcher Korpora. Danach erklären wir die Erstellung von zwei Korpora: Ein Korpus beinhaltet manuelle Annotationen gemäß einem neuen Annotationsschema für sechs Kategorien von textuellen semantischen Relationen, der andere Korpus wurde mithilfe von Schwarmauslagerung (Crowd-Sourcing) erstellt, welches Daten aus dem Internet sammelt.

Danach wird die Erkennung von textueller Verwandtheit (textual relatedness) vorgestellt. Obwohl *Relatedness* normalerweise benutzer- und situationsabhängig ist, kann es in der Praxis helfen, problematische Fälle auszusortieren. Es ist linguistisch indiziert und ist ein schwächeres Konzept als semantische Ähnlichkeit. In Experimenten zeigen wir, dass ein Alignierungsmodell, das auf Prädikat-Argument-Strukturen basiert und dabei Relatedness als Maß benutzt, einem RTE-System helfen kann, diejenigen Fälle (UNKNOWN) zu erkennen, die weder als Folgerung (ENTAILMENT) noch Widerspruch (CONTRADICTION) zu kategorisieren sind und außerdem auch zur Verbesserung der Gesamtleistung in der RTE-Aufgabe mit drei Antworten beiträgt.

Am Ende wird die TSR-Klassifizierung vorgestellt. Wir präsentieren eine Verallgemeinerung von allen vorher beschriebenen Bedeutungsrepräsentationen und stellen einen multidimensionalen Ansatz zur Klassifizierungs vor. Die drei Dimensionen dieses Ansatzes sind neben Verwandtheit (*Relatedness*), auch Inkonsistenz (*Inconsistency*) und Ungleichheit (*Inequality*). Dieser Ansatz wird mit verschiedenen Korpora evaluiert und es wird deutlich, dass dies eine allgemeine Lösungsmöglichkeit für Folgerungserkennung (RTE), Identifizierung von Paraphrasen und anderen TSR-Erkennungsaufgaben ist. Die Performanz des implemtierten Systems ist auf derselben Stufe wie die der andereren Systeme.

Die Arbeit schließt ab mit einem Blick auf mögliche zukünftige Er-

weiterungen der vorgestellten Ansätze. Einige der beschriebenen Modules des Gesamtsystems wurden schon erfolgreich auf andere Probleme der natürlichen Sprach-verarbeitung angewandt. Diese positiven Ergebnisse bestätigen diese Forschungs-richtung und erweitern das Anwendungsgebiet.

Acknowledgements

In retrospect, the past three years forms the best time of my life till now, which (accidentally) includes my pursuit of the PhD degree. I am deeply grateful to many people being around or geographically far away for accompanying with me. I am glad to take this opportunity to mention their names, express my gratitude, and share my happiness with them.

If I am allowed, I would like to start with the farthest ones, my parents, WANG Dingzhu and ZHU Qunhuan, and my wife, NG Manwai. In fact, I am quite sure that they know absolutely nothing about the content of my work, but they still unconditionally and continuously support me to pursue whatever I want. I still owe them a Chinese translation of the dissertation, which my father once asked me for. In addition, according to this study[1], the probability of obtaining the doctor degree is higher, after getting married.

My first (research-oriented) thanks go to my supervisor, Hans Uszkoreit. He is such a great supervisor for me that I have plenty of freedom to choose the research topics to work on, the approaches to solve the problem, and the time to finish. The advices from him keep me as an idealist in thinking while a practitioner in action, which has a profound impact on my way of doing research. Besides, he is such a reliable person that he always has some solution for my problems, provided that he replies to my email.

Another great thank to my main collaborator, ZHANG Yi. He is an extremely nice person to work with, since usually I only need to discuss something with him and he will do it soon. We had various enjoyable brainstorming discussions, paper writing, poster drawing, and deadline catching. Many thanks to my advisor in the partner university, Johns Hopkins University, Chris Callison-Burch. He is probably one of the most friendly persons I have ever met. He kindly supervised me during my exchange period and gave me many insights of the field. Another thank goes to my former supervisor in Shanghai Jiao Tong University, YAO Tianfang, who invited me for a two-week visit and I enjoyed my pleasant stay back to my former lab.

Many thanks to Günter Neumann, who participated in the RTE challenges with me and we always looked forward to the brighter future together. Many thanks to Caroline Sporleder. Inter-annotator commu-

---

[1]http://www.phdcomics.com/comics/archive.php?comicid=1381

tation was really a nice experience to me. I would also like to thank those people with whom I had inspiring discussions (or bothered): Dietrich Klakow, who "enjoyably" reviewed my dissertation; Manfred Pinkal, who encouragingly pointed out some errors; Alexander Koller, who politely reminded me of some related work; Ido Dagan, Bernardo Magnini, and Sebastian Padó, from whom I always obtain (steal?) some ideas, after talking with them. I really enjoy collaborating and discussing with them and expect more in the future.

I sincerely thank all the people I met in Johns Hopkins University, in particular, Jason Eisner (leading the reading group), Sanjeev Khudanpur (helping me with the structure of my dissertation), Mark Dredze (teaching a machine learning course), and colleagues who made my research life in the States much easier, Anoop Deoras, Markus Dreyer, Alexandre Klementiev, LI Zhifei, Carolina Parada, Delip Rao, Jason Smith, WANG Ziyuan, XU Puyang, Omar Zaidan, and ZHOU Haolang. There are many people I met and talked to in various conferences, workshops, and meetings, whom I cannot enumerate all the names here. I would like to say "thank YOU".

I also want to express my gratitude (and sorriness) to those people who helped me to proof-read my dissertation. I guess it was really a hard time for them to find a polite way to say "this is totally bullshit". So after some hesitation, they all started with "it's good", which psychologically encouraged me to read my dissertation again and again. Without their help, it is much less readable, so my appreciation is not just for politeness. In particular, Grzegorz Chrupała, Bart Cramer, Rebecca Dridan, Hagen Fürstenau, Konstantina Garoufi, LI Linlin, Alexis Palmer, Caroline Sporleder, and Sabrina Wilske helped me to read individual chapter(s), and Günter Neumann, Hans Uszkoreit, and ZHANG Yi read the whole dissertation. Special thanks go to Sabrina Wilske, who helped me to translate the abstract into *Zusammenfassung*. All the remaining errors (including the German ones) are certainly my own fault.

For the non-research part of my life, I would like to thank all my friends (not limited to those having meals or coffee breaks with me): Lee Lap-Kei, QU Lizhen, SUN He, SUN Weiwei, WANG Yafang, XU Jia, XU Zenglin, YANG Bin, YE Min, and those friends participated in the "Mars' hat" project: CHEN Yu, Grzegorz Chrupała, Georgiana Dinu, Antske Fokkens, Konstantina Garoufi, LI Linlin, NG Manwai, WEI Shuai, Sabrina Wilske, ZHANG Yajing, and ZHANG Yi.

Last but not least, I deeply thank my scholarship program, IRTG/PIRE,

It is always nice to see an end, as it entails another start.

Contents

List of Figures

List of Tables

1 Introduction

This chapter gives an overview of this dissertation. We start with an introduction of recognizing textual entailment (RTE). For natural language processing (NLP), especially for natural language understanding, this is a useful and challenging task. Then we define the scope of the task under consideration in this dissertation. Following that, a proposal is presented, which provides two categories of approaches, *intrinsic* and *extrinsic* ones. In the end, the structure of the dissertation is given as well as a summary of each chapter.

1.1   Motivation

Entailment is widely used in many aspects of the human life. Assume that someone is seeking for something and he or she searches for the answer from books, friends, or the Web. In most cases, the information gathered or retrieved is not the exact answer, although the (information) seeker may have one in his or her mind. Instead, the consequences of the original goal may be detected, so the *inference* plays a role and confirms or denies the original information being sought.

For instance, John wants to know whether the Amazon river is the longest river in the world. Naturally, he can find the exact lengths of the Amazon and other rivers he knows of, and then compare them. But once he sees "Egypt is one of the countries along the longest river on earth", he can already infer that Amazon is not the longest river, since Egypt and the Amazon river are not on the same continent. Similarly, assuming that Albert is not sure who is the current president of the U.S., Bush or Obama, since both "president Bush" and "president Obama" are retrieved. If he performs an inference based on one of the retrieved documents containing "George Bush in retirement", the answer is obvious. In short, finding out the exact information is not always trivial, but inference can help a lot. In both cases, the retrieved information *entails* the answer instead of being the precise answer.

Entailment also occurs frequently in our daily communication, with respect to language understanding and generation. Usually we do not literally interpret each other's utterances, nor express ourselves in a straight way. For example,

- *Tom: Have you seen my iPad?*

- *Robin: Oh, nice! I'd like to have one too.*

- *Tom: You have to get one.*

The dialogues seem to be incoherent, if we literally and individually interpret each sentence. Firstly, Tom asks a yes-no question, but Robin does not directly give the answer. Instead, Robin implies that he has not seen it before the conversation by showing his compliment to it ("Oh, nice!"). Probably Tom is showing his iPad to Robin during the conversation. Robin's second sentence also implies that he does not have an iPad till then, and therefore Tom's response is a suggestion for him to get one.

If we literally interpret the conversation, it sounds a bit awkward. Here is one possibility:

- *Tom: Here is my iPad.*

- *Robin: I haven't seen it before. It is nice. I don't have one, but I'd like to have one.*

- *Tom: I suggest you get one.*

Although the interpreted version may be easier for the computers to process human dialogues, the original conversation occurs more naturally in our daily life. Each utterance in the interpreted version is actually implied or *entailed* by the utterances in the original conversation. Consequently, if we want to build a dialogue system, dealing with this kind of implication or entailment is one of the key challenges. Let alone there is common sense knowledge which does not appear in the dialogue but is nevertheless acknowledged by both speakers, e.g., what an iPad is.

In general, following Chierchia and McConnell-Ginet (2000), we refer to such a relationship between two texts as *Textual Entailment* in this dissertation. The task, *Recognizing Textual Entailment* (RTE), is a shared task proposed by Dagan et al. (2006), which requires the participating systems to predict whether there exists a textual entailment relation between two given texts, usually denoted as *text* (**T**) and *hypothesis* (**H**). An example is like this:

T: *Google files for its long awaited IPO.*

H: *Google goes public.*

As for the NLP perspective, RTE can be viewed as a generic semantic processing module, which serves for other tasks. For instance, it has already been successfully used for question answering (Harabagiu and Hickl, 2006), including answer validation (Peñas et al., 2007, Rodrigo et al., 2008), information extraction (Roth et al., 2009), and machine translation evaluation (Padó et al., 2009a). In the long term, RTE can also play an important role in understanding conversation dialogues (Zhang and Chai, 2010), metaphors (Agerri, 2008), and even human-robot communication (Bos and Oka, 2007).

1.2   Scope

Textual entailment originates from *entailment* or *logical implication* in logic. Typically it is defined in terms of necessary truth preservation, which is some set of sentences T entails a sentence A if and only if it is necessary that A be true whenever each member of T is true. It can be represented as $A \Rightarrow B$ or $A \subseteq B$. Notice that we only consider the cases when A is true, excluding the $\neg A$ cases. In linguistics, this phenomenon is similar to *implication*, which includes conventional and conversational implicature as well. For instance, the "Google" example shown at the end of last subsection is a conventional implicature.

Modality is another issue to mention. In the most common interpretation of modal logic, people consider "logically possible worlds" (Lewis, 1986). One proposition is a

- *Necessary* or *Impossible* proposition: if a statement is true or false in all possible worlds;

- *True* or *False* proposition: if a statement is true or false in the actual world;

- *Contingent* proposition: if a statement is true in some possible worlds, but false in others;

- *Possible* proposition: if a statement is true in at least one possible world.

Ideally, if the entailment relation holds between two propositions, it holds in all possible worlds; while in practice, the language usually concerns a subset of all the possible worlds. The simplest case would be the actual world, if the modality does not change. Therefore, we can group all the text pairs into two categories:

1. The speaker does not change the modality;

2. The speaker changes the modality into some other possible world(s) or even all the possible worlds (e.g., universal quantifiers).

Since we cannot really verify the relation in *all* possible worlds, our goal here is to know whether it holds in the possible worlds set by the context. Furthermore, in practice, most of the work in this dissertation focuses on the first category, which can roughly be viewed as the actual

world; and the others allow different possible worlds, e.g., entailment involving temporal expressions (Chapter 4).

The work discussed in this dissertation differs from traditional approaches to solving logical entailment in the following two ways: a) we make the simplifying assumptions discussed above; and b) instead of dealing with propositions or logic forms, we handle plain texts, which leads us to face the ambiguous nature of natural languages.

If we make an analogy to the "triangle" in the machine translation (MT) community (Figure 1.1), we can visualize the RTE task as a rectangle (Figure 1.2). The MT triangle says, from the source language to the target language, there exist many possible paths. We can do a direct translation based on the surface strings, or we can apply some linguistic analysis first to obtain the meaning of the two texts. Furthermore, the depth of this analysis is underspecified, and thus, the representation of the (approximated) meaning varies. Similarly, the RTE rectangle does not require an explicit or concured meaning representation. The key criterion is to verify the inclusion or subsumption relation between the two sides.
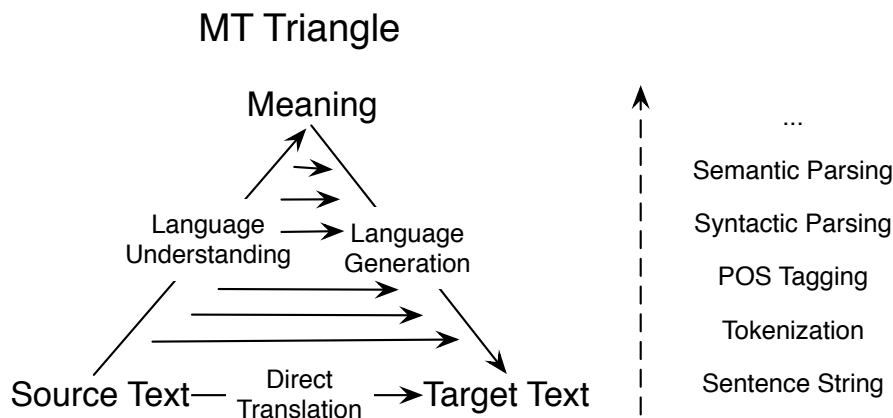
Figure 1.1: The MT triangle

Besides the common features, several differences are noticeable as well:

1. In MT, the source text is given, but the target text is not; while in RTE, both texts are given.

2. In MT, the source text and the target text are in different languages (otherwise, it is a monolingual paraphrase generation system instead of an MT system); while in RTE, the two texts are in the same

language[1].

3. In MT, the two texts share a single meaning; while in RTE, there is an inclusion between the meaning of the two texts (even at the "deepest" level, if possible). In other words, if we have the full meaning representation of the text in MT, we need no transfer rules; while in RTE, there must be a process of comparing the two structures derived from the texts[2]. And this makes the different shapes of the two models.
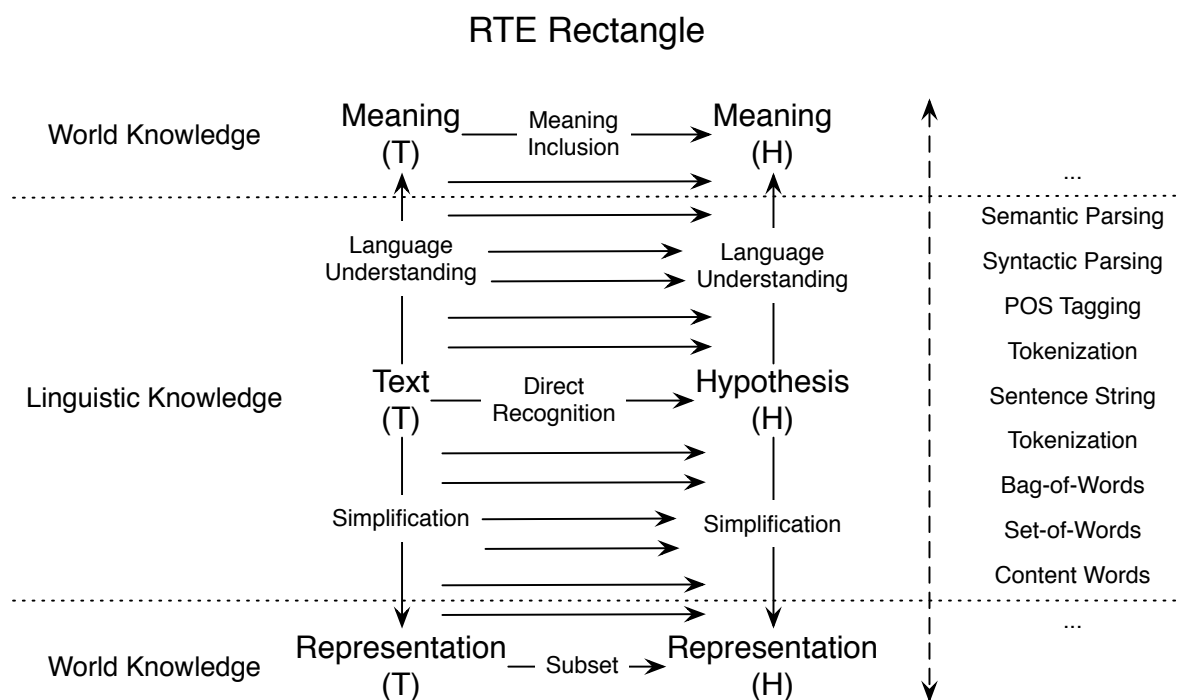
Figure 1.2: The RTE rectangle

The two dimensions in Figure 1.2 exactly describe the key issues involved in the RTE task[3]:

- What is a proper (meaning) representation? Or how "deep" should we analyze the text?

- How can we detect such entailment relations between two texts?

---

[1]Some recent research focuses on cross-lingual textual entailment (Mehdad et al., 2010), where they investigate an entailment relation between two texts in different languages.

[2]Some other researchers might not agree on this. Translation may not just preserve the *meaning* but the *mental status*. Nevertheless, this issue becomes more severe in RTE.

[3]In fact, this also influences the architecture design of the RTE systems, which we see more in Section 2.

Our consideration of this task is also around these two questions. Thus, in this dissertation, we discuss the following aspects: meaning representation, different entailments, external knowledge, semantic relations, and text granularity.

Meaning Representation are mentioned several times, for instance, in Section 2.2, Section 5.4.2, Section 8.1, and Section 9.1. In this work we focus primarily on meaning representations based on dependency relations between words and/or tokens. In some cases, meaning is represented at the syntactic level, and in others at the semantic level. Another variation is whether we use tree structures or graphs to represent meaning. We call them all as *meaning representation* in general, although most of them only approximate the full meaning.

In fact, one of the main motivations of the RTE task is to seek alternative ways to do inference, other than to access the full meaning of the text. In this sense, the plain text itself can be viewed as one meaning representation, and the enrichment or abstraction of the structural information provides other options (Figure 1.2). Instead of performing reasoning on the full meaning, the inferences can be done on all these different levels of representations.

Different Entailments can be viewed as a classification or categorization of different cases of entailment. In logic, the notion of entailment is clearly defined and strict; while in computational linguistics, textual entailment more or less takes the range of implication. Therefore, Section 3.1 shows the complexity of this phenomenon, and both Chapter 4 and Chapter 5 deal with subsets of the problem. Two approaches show different degrees of the abstraction of the (inference) rules, which can be a direct textual (or dependency-relation-based) rule application (Chapter 5) or a more abstract rule representation (Chapter 4).

External Knowledge is another interesting issue to investigate. Section 2.1 includes an overview of resources used in the RTE community. According to the original RTE proposal (Dagan et al., 2006), the policy of using external knowledge is that **H** should not be validated by the external knowledge alone, regardless of the given **T**, e.g., searching **H** on the Web.

Although most of our work focuses on the information contained within

the texts, Chapter 5 is about applying an external inference rule collection to the RTE task. In other sections, like Section 4.4 and Section 8.2, we also make use of external lexical semantic resources. However, notice that in many cases it is also difficult to draw a clear boundary between the *linguistic* meaning contained in the given texts and world knowledge from outside.

Semantic Relations   between two texts are the superset of the entailment relation. Besides the entailment relation, there are other possible semantic relations, e.g., equivalence (i.e., bi-directional entailment), contradiction, etc. In tasks like paraphrase acquisition and natural language generation, the directional entailment test is not enough. Instead, an equivalence test has to be performed. While in other tasks like information verification and sentiment analysis, contradictory or contrasting information is of great interest.

We show the advantage of tackling multiple relations simultaneously, as the search space for each task is largely reduced due to this kind of "filtering" (Chapter 9). The upper bound of such relations is a pair of identical texts, and the lower bound is a random pair of texts, which are independent of each other. Section 2.4 discusses the related work, and Section 6.1 and Chapter 9 focus on this generalized problem.

Text Granularity   should also be mentioned here. In the scope of this dissertation, we mainly work with pairs of text, and each text consists of one or more sentences. We assume they together provide a certain context or possible world, where the relationship between them is invariant. The granularity is also the main difference between the traditional lexical semantic relations (like synonym, hypernym, etc.) and the *textual* semantic relations we deal with. For instance, as a single term, "on this Wednesday" entails "in this week", while the proposition "I'm not working on this Wednesday" does not entail "I'm not working in this week". The monotonicity cannot be always preserved. Therefore, many issues discussed in the lexical semantics (e.g., privative adjectives) are not the main focus of this dissertation, where we rely more on the external knowledge resources.

1.3 Proposal

To tackle this problem, we look at it from two different angles, *intrinsically* and *extrinsically*:

- **Intrinsically**, we use specialized RTE modules to tackle different cases of entailment.

- **Extrinsically**, we put the entailment relation into a more general framework, i.e., textual semantic relations.

In particular, due to the complexity of the problem, we propose an extensible architecture with different specialized modules to handle different cases of textual entailment in parallel (Chapter 3). For instance, we develop a module especially for dealing with those entailments where temporal reasoning is involved (Chapter 4). This can be further extended into reasoning or resolution among other named-entity types like location names, person names, and organization names (Section 4.7). The key requirement for a "good" module is that it should be precision-oriented, which is different from the recall-oriented pipeline architecture.

The concept of "module" can be further generalized into "resource". Once a subset of entailments can be solved by one specific resource or external knowledge base, we develop a "module" based on it. For example, we apply an inference rule collection to entailment recognition and also treat it as a specialized module dealing with a target subset, i.e., those cases that can be solved or at least covered by the rules (Chapter 5).

These methods are all based on the assumption that we can decompose the text into smaller units, which are semantically atomic (for that approach). When we use temporal reasoning, person name resolution, or inference rules, we put emphasis on some of the units, namely temporal expressions, person names, and those parts covered by the rules. In practice, one semantic unit can also be realized as a logic proposition, a predicate-argument pair, a syntactic dependency triple, or even a single word. Section 9.1 gives a generalized form for all the representations we have utilized in our work. Based on this unified framework, extra modules can be easily incorporated into the architecture.

Apart from tackling RTE in a standalone manner, we also look at other relevant relations between texts. We firstly construct two corpora for the evaluation of our developed system(s) (Chapter 7). We design a new annotation scheme of six categories of textual semantic relations and

manually annotate a corpus (Section 7.2). We also make use of the crowd-sourcing technique to collect more data from the Web (Section 7.3).

Then, we propose an intermediate step before entailment recognition, which is to recognize *textual relatedness* (Chapter 8). We further extend the method, incorporating two extra measurements, *inconsistency* and *inequality*. Four textual semantic relations, PARAPHRASE, ENTAILMENT, CONTRADICTION, and UNKNOWN, can thus be classified by this multi-dimensional approach (Chapter 9). Experiment results show that 1) filtering out other possible relations can reduce the search space for entailment recognition; and in the meantime, 2) multiple semantic relations can be recognized simultaneously.

As the original motivation to propose RTE is to build a unified semantic interface for NLP tasks like information extraction, question answering, summarization, etc. (Dagan et al., 2006), it is worthwhile to see the (dis)similarity between RTE and other semantic relations or NLP tasks, and our work is in the right direction to achieve that goal.

1.4   Highlights

- An extensible architecture with specialized modules for recognizing textual entailment;

- A general framework for textual semantic relation recognition;

- Construction of two heterogeneous corpora with different methodologies;

- Comparison of different depths of linguistic processing and various resources;

- Comparison of rule-based methods and statistical methods.

1.5   Organization

Figure 1.3 shows the structure of the dissertation, and we briefly introduce each chapter in the following:

- Chapter 2: We present a summary of the previous work done by other researchers and the relation to this dissertation, including available resources, meaning representation derivation, entailment recognition,

as well as other related tasks such as paraphrase acquisition. We also show the state-of-the-art system performance and their application to other NLP tasks.

Part A: Intrinsic Approaches

- Chapter 3: This chapter is the overview of the next two chapters. We introduce the extensible architecture of our (intrinsic) approach to the RTE task with specialized modules handling different cases of entailment. We also mention some possible extensions of the approach, as well as some related work done by other researchers.

- Chapter 4: We start with one specialized module for tackling textual entailment pairs with temporal expressions. A separate Time Anchoring Component (TAC) is developed to recognize and normalize the temporal expressions contained in the texts. We then show that the generalization of this module can handle texts containing other types of named-entities as well. The experimental results show the advantages of the precision-oriented specialized entailment modules and suggest a further integration into a larger framework for general textual inference systems.

- Chapter 5: This chapter is mainly about applying external knowledge bases to the RTE task. We extend and refine an existing inference rule collection using a hand-crafted lexical resource. The experimental results demonstrate that this is another precision-oriented approach, which can also be viewed as a specialized module. As alternative resources, we also present a pilot study on acquiring paraphrased fragment pairs in an unsupervised manner.

Part B: Extrinsic Approaches

- Chapter 6: This chapter introduces the second part of the dissertation. Basically, instead of tackling the RTE task in a standalone manner, we are looking for its connection to other tasks, i.e., to recognize other semantic relations between texts. We firstly describe the motivation for making this generalization and then present a framework for handling all these relations simultaneously.

    – Chapter 7: This chapter is about the corpora used in this dissertation. We firstly give an overview of all the datasets we have, followed by a discussion of the methodologies used in their construction. Then we elaborate on two corpora we constructed: one has a new annotation scheme of six categories of textual semantic relations with manual annotations; and the other uses a crowd-sourcing technique to collect data from the Web.

    – Chapter 8: We focus on textual relatedness recognition in this chapter. Although *relatedness* is usually user-dependent and situtation-dependent, in practice, it can help to filter out noisy cases. It is linguistically-indicated and can be viewed as a weaker concept than semantic similarity. In the experiments, we show that an alignment model based on predicate-argument structures using this relatedness measurement can help an RTE system to recognize the UNKNOWN cases at the first stage. Further, it can contribute to the improvement of the system's overall performance as well.

    – Chapter 9: Finally, we present the work on textual semantic relation (TSR) recognition. We start with a generalization of all the meaning representations described in the previous chapters. Then, a multi-dimensional classification approach is introduced, including *relatedness* as one of the dimensions. The other two dimensions are *inconsistency* and *inequality*. We evaluate our approach on the datasets described in Chapter 7 and show that this is a generalized approach to handle entailment recognition, paraphrase identification, and other textual semantic relation recognition tasks.

- Chapter 10: We summarize the dissertation and recapitulate the issues. Several open questions in RTE are discussed, and several applications to other NLP tasks are shown, where RTE is used as a valuable component. Possible directions for future exploration are also pointed out.
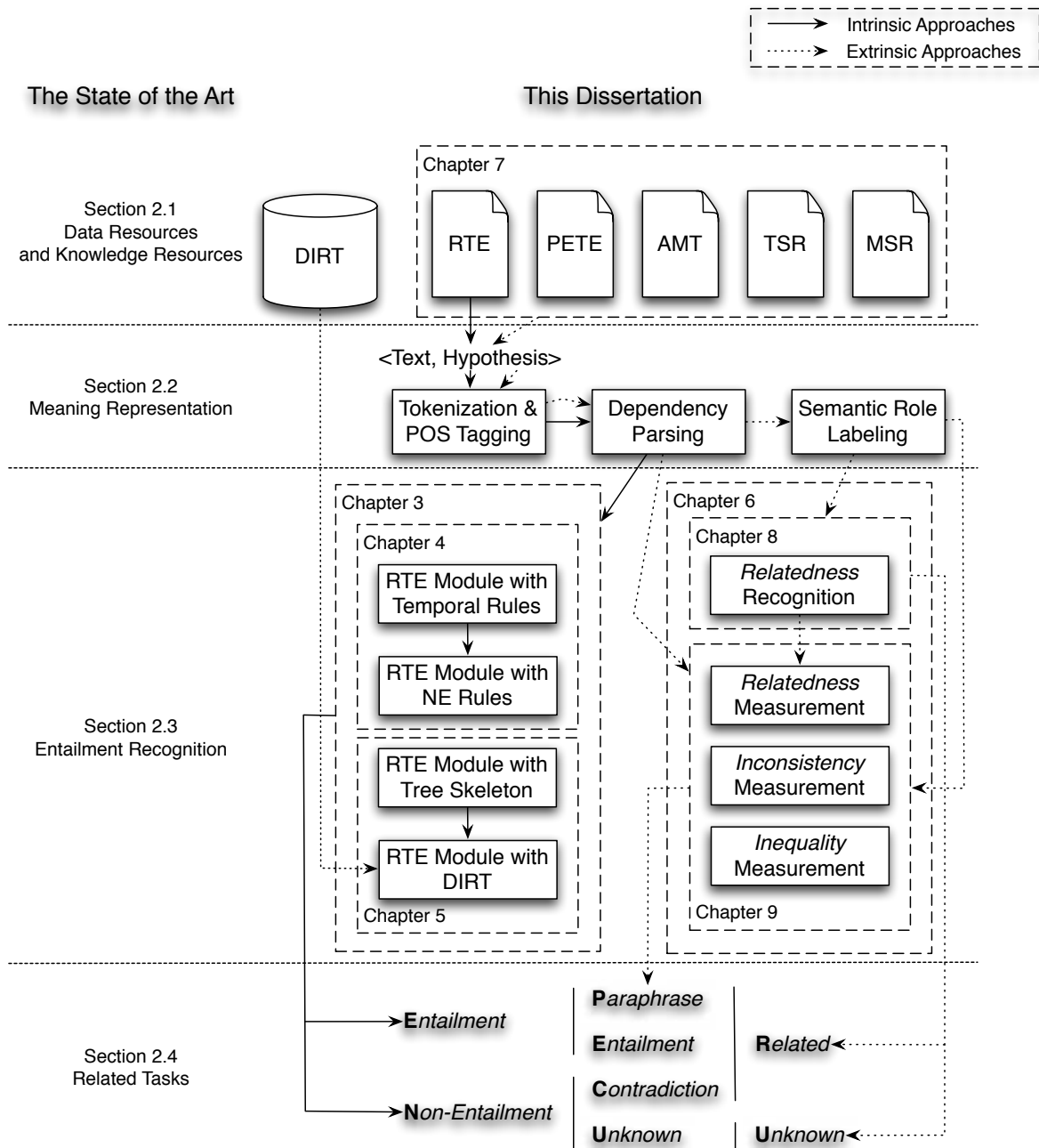
Figure 1.3: Organization of the dissertation

2 The State of the Art

This chapter is mainly about the related work done by others and their relation to this dissertation. We firstly introduce the resources used in the community, including the datasets and annotations (Section 2.1). Following that, the common approaches to preprocessing the natural language text (Section 2.2) and recognizing the entailment relation between two texts (Section 2.3) are described. Section 2.4 introduces some other tasks related to entailment recognition and Section 2.5 discusses the state-of-the-art system performance and applications to other NLP tasks.

2.1   Data Resources and Knowledge Resources

In this section, we start with existing datasets as well as manual annotations on them. Then we focus on two categories of knowledge resources applied in most of the RTE systems, general-purpose lexical semantic resources and textual inference rule collections. Most of the resources discussed here can be easily found and freely used for research purposes[1].

2.1.1   Datasets and Annotations

One large collection is provided by the *Recognizing Textual Entailment* (RTE) community, following each year's challenge, from RTE-1 in 2005 (Dagan et al., 2006) till RTE-5 in 2009 (Bentivogli et al., 2009). The corpora from the first two RTE challenges are annotated with two labels: One is YES, meaning that there is an entailment relation from the first text, *text* (**T**), to the second text, *hypothesis* (**H**); and the other label is NO, meaning there is no such relation. Starting from the RTE-3 Pilot task,[2] the annotation is extended to three labels, ENTAILMENT, CONTRADICTION, and UNKNOWN. ENTAILMENT is the same as the previous YES; but NO is divided into CONTRADICTION and UNKNOWN, to differentiate cases where **T** and **H** are contradictory to each other from all the other cases. The RTE data are acquired from other NLP tasks, like information retrieval, question answering, summarization, etc., and thus, in some sense, the corpora construction is more application-driven than linguistically motivated.

Besides the gold-standard labels from the RTE challenges, some researchers also made efforts to enrich the annotations by adding more linguistic analyses. For instance, Garoufi (2007) proposed a scheme for annotating **T**-**H** pairs, which models a range of diverse entailment mechanisms. There was an inventory of 23 linguistic features, including *acronym*, *hypernym*, *apposition*, *passivization*, *nominal*, *modifier*, and so on. They annotated a considerable portion of the RTE-2 dataset (400 positive **T**-**H** pairs) and examined from various aspects the performance of the RTE systems participating in the RTE-2 Challenge (Bar-Haim et al., 2006). Sammons et al. (2010) also argue that the single global

---

[1]http://www.aclweb.org/aclwiki/index.php?title=Textual_Entailment_Resource_Pool

[2]http://nlp.stanford.edu/RTE3-pilot/

label with which RTE examples are annotated is insufficient to effectively evaluate RTE system performance and more detailed annotation and evaluation are needed. They used insights from successful RTE systems to propose a model for identifying and annotating textual inference phenomena in textual entailment examples, and they presented the results of a pilot annotation study that showed this model was feasible and the results immediately useful.

More research focused on a subset of the entailment phenomena. The Boeing-Princeton-ISI (BPI) textual entailment test suite[3] was specifically designed to look at entailment problems requiring world knowledge. It contains 125 positive and 125 negative (no entailment) pairs. Compared with the PASCAL RTE data sets, the BPI suite is syntactically simpler but semantically challenging, with the intension of focusing more on the knowledge rather than just linguistic requirements. In particular, the examples include inferences requiring world knowledge, not just syntactic manipulation. An analysis of what kinds of knowledge are required for the 125 positive entailments was also performed, resulting in 15 somewhat loose categories of knowledge.

Mirkin et al. (2010b) performed an in-depth analysis of the relation between discourse references and textual entailment. They identified a set of limitations common to the handling of discourse relations in virtually all entailment systems. Their manual analysis of the RTE-5 dataset (Bentivogli et al., 2009) shows that while the majority of discourse references that affect inference are nominal coreference relations, another substantial part is made up by verbal terms and bridging relations. Furthermore, they demonstrated that substitution alone is insufficient for the resolution of discourse references and it should be tightly integrated into entailment systems instead of being treated as a preprocessing step. In addition, their analyses also suggest that in the context of deciding textual entailment, reference resolution and entailment knowledge can be seen as complementary ways of achieving the same goal, namely enriching **T** with additional knowledge to allow the inference of **H**. Given that both of the technologies were still imperfect, they envisaged the way forward as a joint strategy, where reference resolution and entailment rules mutually filled each others gaps.

In RTE-4 (Giampiccolo et al., 2009), Wang and Neumann (2009) proposed a novel RTE system architecture, which consists of specialized

---

[3]`http://www.cs.utexas.edu/~pclark/bpi-test-suite/`

modules dealing with different types of entailment (more details can be found in Chapter 3). This was confirmed by other researchers as well. Bentivogli et al. (2010) proposed a methodology for the creation of specialized data sets for textual entailment, made of monothematic **T-H** pairs (i.e., pairs in which only one linguistic phenomenon relevant to the entailment relation is highlighted and isolated). They carried out a pilot study applying such a methodology to a sample of 90 pairs extracted from the RTE-5 data and they demonstrated the feasibility of the task, both in terms of quality of the new pairs created and of time and effort required. The result of their study is a new resource that can be used for training RTE systems on specific linguistic phenomena relevant to inference.

So far, we have not touched the issue of data collection, which we leave for later (Chapter 7). The common source of the RTE data is other NLP tasks, e.g., information extraction, summarization, etc. Alternative inexpensive ways of corpora construction are worth investigating as well (Wang and Callison-Burch, 2010).

Apart from the entailment-centered datasets, there are also corpora containing more semantic phenomena. One early related work was done by Cooper et al. (1996), and they named the corpus FraCaS (a framework for computational semantics). They focused more on the linguistic side, aiming to cover different linguistic/semantic phenomena. The annotation is similar to the three-way RTE. However, this dataset was manually constructed and the sentences were carefully selected. It turned out to have a "text-book" style, which is quite different from the real data we usually need to process. The size of the dataset is also far from enough for training a robust machine-learning-based RTE system.

2.1.2 General Knowledge Bases

In the recent RTE challenges, submitted systems are also required to provide ablation test results by excluding the external knowledge bases one by one. Therefore, the impact of each resource can be easily seen.

In both RTE-4 (Giampiccolo et al., 2009) and RTE-5 (Bentivogli et al., 2009), three categories of resources are widely used:

- WordNet (Fellbaum, 1998) and its extensions: they are used in order to obtain synonyms, hyponyms, and other lexically related terms.

- VerbOcean[4] (Chklovski and Pantel, 2004) and DIRT (Lin and Pantel, 2001)[5]: they are mostly used in order to obtain relations between verbs or predicates.

- Wikipedia[6], and other gazetteers: they are used to recognize and resolve the named-entities.

WordNet is widely used in almost all the RTE systems. The most common usage is to compute a similarity score between two words using the semantic links, e.g., synonyms, hyponym/hypernyms, etc. Galanis and Malakasiotis (2009) and Malakasiotis (2009) experimented with a list of similarity measurements, including Cosine similarity, Euclidean distance, Levenshtein distance, and so on. Clark and Harrison (2009a,b) utilized WordNet to improve the robustness of the logic inference by enlarging the coverage and from 4% to 6% accuracy on the final result was attributed to it. However, on average, among the 19 participating systems of RTE-5, only 9 of them found WordNet effective, 7 of them found it harmful to the final result, and 3 observed no effects. It seems that an appropriate usage of such general-purpose resources still needs further exploration.

Balahur et al. (2009) and Ferrández et al. (2009) used VerbOcean and VerbNet[7] (Kipper et al., 2006) to capture relations between verbs. Two verbs were related if they belonged to the same VerbNet class or a subclass of their classes; or they had one of the VerbOcean relations: *similarity*, *strength*, or *happens-before*. Mehdad et al. (2009b) made use of VerbOcean in a similar manner. The difference was that they transformed the verb relations into rules and assigned different weights to the rules based on an editing distance model.

As for Wikipedia, Shnarch (2008) created an extensive resource of lexical entailment rules from Wikipedia, using several extraction methods. It consisted of 8 million rules, and was found to be fairly accurate. Bar-Haim et al. (2009) incorporated those rules in their system. Li et al. (2009b,a) used Wikpedia mainly for named-entity resolution, since there are different references to the same entity. They combined the information from Wikipedia with outputs of other modules and constructed graphs of entities and relations for further processing. Both Mehdad

---

[4]`http://demo.patrickpantel.com/demos/verbocean/`

[5]We focus more on the lexical resources in this subsection, and leave textual inference rules for the next subsection.

[6]`http://www.wikipedia.org/`

[7]`http://verbs.colorado.edu/~mpalmer/projects/verbnet.html`

et al. (2009b) and Mehdad et al. (2009a) treated Wikipedia as an alternative source for lexical similarity measurement, while the former used an editing distance model and the latter a kernel-based method.

As one of the top systems, Iftene (2009) and Iftene and Moruz (2009) incorporated all these resources and confirmed the contribution of each one via ablation tests. In addition to the widely used resources, Nielsen et al. (2009) took Propbank (Palmer et al., 2005) to help them with obtaining their facet-based representation, and Ferrández et al. (2009) defined one similarity score based on FrameNet (Baker et al., 1998).

Besides the ablation tests of the participating systems, Mirkin et al. (2009a) studied the evaluation methods for the utility of lexical-semantic resources on the RTE task. They proposed system- and application-independent evaluation and analysis methodologies for resource performance, and systematically applied them to seven prominent resources, including WordNet and Wikipedia. Their evaluation and analysis provide a first quantitative comparative assessment of the isolated utility of a range of prominent resources for entailment rules. In Section 8.3.4 we also compare several lexical resources to see their impact on relatedness recognition as well as entailment recognition.

In this subsection, we cannot cover all the general knowledge resources used by the RTE systems. A more thorough survey can be found in overview papers of RTE-4 (Giampiccolo et al., 2009) and RTE-5 (Bentivogli et al., 2009).

2.1.3   Textual Inference Rules

In contrast to the widely used lexical resources, the usage of paraphrase collections, or automatic acquisition of paraphrases is restricted to a small number of systems. A number of systems used hand crafted rules, e.g., Bos and Markert (2006), but their number did not get close to the level of coverage needed. An alternative to the logic inference rule is the textual inference rule.

We use a liberal definition of textual inference rules here. Basically, we mean automatically acquired rewriting rules in other representations than the logic form. A number of such inference rule/paraphrase collections are available (Szpektor et al., 2004, Sekine, 2005). We focus on one representative and widely-used one, the DIRT collection (Lin and Pantel, 2001). The acquisition algorithm has been introduced by Lin and Pantel

$$\frac{X \text{ put emphasis on } Y}{}$$
$$\approx X \text{ pay attention to } Y$$
$$\approx X \text{ attach importance to } Y$$
$$\approx X \text{ increase spending on } Y$$
$$\approx X \text{ place emphasis on } Y$$
$$\approx Y \text{ priority of } X$$
$$\approx X \text{ focus on } Y$$

Table 2.1: Examples of the DIRT algorithm output, most confident paraphrases of *X put emphasis on Y*

(2001) and it is based on what is called the *Extended Distributional Hypothesis*. The original Distributional Hypothesis (DH) states that *words* occurring in similar contexts have similar meaning, whereas the extended version hypothesizes that *phrases* occurring in similar contexts are similar.

An inference rule in DIRT is a pair of directional relations between two text patterns with variables (Szpektor et al., 2007). The left-hand-side pattern is assumed to entail the right-hand-side pattern in certain contexts, under the same variable instantiation. The definition relaxes the intuition of inference, as the entailment is only required to hold in *some* but not *all* contexts, motivated by the fact that such inferences occur often in natural text. Table 2.1 gives a few examples of rules contained in DIRT.

There are also other inference rule collections in similar style. For example, unlike most work on unsupervised entailment rule acquisition which focused on rules between templates with two variables, Szpektor and Dagan (2008) investigated two approaches for unsupervised learning of *unary* rules, i.e., entailment rules between templates with a single variable, and outperformed the proposed methods with a binary rule learning method. The first approach was based on distributional similarity measures and the second approach derived unary rules from a given database of binary rules. They tested the different approaches utilizing a standard IE test-set and their results suggest the advantage of learning unary rules: (a) unary rule-bases perform better than binary rules; (b) it is better to directly learn unary rules than to derive them from binary rule-bases.

Instead of learning the rules from corpora, Aharon et al. (2010) generated inference rules between predicates solely from the information con-

tained in FrameNet. They showed that the resulting rule-set largely complemented the rules generated from WordNet, because it contained argument mappings between non-substitutable predicates, which are missing from WordNet, as well as lexical relations that are not included in WordNet. They also pointed out that combining FrameNet and WordNet rule-sets in a transitive manner instead of their union was worth investigating in the future. In fact, similar treatment is made when we combine different lexical resources (Section 8.3.4).

Apart from enlarging the coverage of the rule-set, another work done by Berant et al. (2010) focused on the accuracy of the rules collected. They defined a graph structure over predicates that represented entailment relations as directed edges, and used a global transitivity constraint on the graph to learn the optimal set of edges. They used Integer Linear Programming to solve the optimization problem and demonstrated empirically that this method outperformed local algorithms as well as a greedy optimization algorithm on the graph learning task. Their global algorithm improved performance by more than 10% over baseline algorithms.

Intuitively such inference rules should be effective for recognizing textual entailment. However, only a small number of systems used DIRT as a resource in the RTE-3 challenge, and the experimental results did not fully show that it has an important contribution. Whereas hand-crafted rules lack coverage, automatically-acquired ones are usually noisy. The details of textual rule application in the RTE systems are discussed in Section 2.3.2.

2.2   Meaning Representation

As we mentioned before, all the approaches dealing with RTE contain two important procedures: meaning representation derivation and entailment relation recognition (Section 1.2). They can be viewed as the vertical and horizontal directions in Figure 1.2 respectively. The meaning representation refers to the representation obtained after the "vertical" processing, i.e., preprocessing (if entailment recognition is treated as the main task). Although these two procedures are intertwined, most of the state-of-the-art systems can be put into this two-staged framework. We discuss the commonly used representations in this section and the methods for entailment recognition in the next section.

Wang (2007) summarized the representations used in the RTE systems participating in RTE-1, RTE-2, and RTE-3. In this section, we focus on the recent trends, namely the participating systems in RTE-4 and RTE-5.

Surface string and bag-of-words representations are the most widely used representations approximating of the meaning of the text. For instance, Galanis and Malakasiotis (2009) used a Maximum Entropy classifier along with string similarity measures applied to several abstractions of the original texts (e.g., the original sentences, the stems of their words, and their POS tags). We also use the bag-of-words representation as one backup strategy to calculate a similarity score between **T** and **H**, since it is simple and robust.

Parsing is also widely considered. While Zanzotto et al. (2009) built their tree kernels on the constituent tree structure, many other systems worked on the dependency trees, e.g., Yatbaz (2009). Unlike two specialized modules proposed by Cabrio et al. (2009) based on the bag-of-words representation, Mehdad et al. (2009b) applied the same EDITS (Edit Distance Textual Entailment Suite) package[8] (Kouylekov and Negri, 2010) on the dependency parse trees. Compared with Galanis and Malakasiotis (2009), Malakasiotis (2009) also incorporated a dependency parser to measure similarity between the grammatical structures of **T** and **H**. Instead of working with the whole dependency tree, Krestel et al. (2009a,b) extracted the (syntactic) predicate-argument structure (PAS), i.e., *subject*, *predicate*, and *object*, and compare two PASes to obtain the similarity score for each **T**-**H** pair. We use such PASes as the meaning representation in one RTE module (Section 5.4.2), but not restricted to (only) *subject* and *object* relations.

Furthermore, Bar-Haim et al. (2009) presented a new data structure, termed compact forest, which allowed efficient generation and representation of entailed consequents, each represented as a parse tree. Rule-based inference was complemented with a new approximate matching measure inspired by tree kernels, which was computed efficiently over compact forests. Using that data structure, they were able to integrate many entailment rules from diverse sources, and showed their contributions to overall performance.

Apart from the structural information of the sentence, named-entities (NEs) are also shown to be important to the RTE task. Castillo and i Alemany (2009) created a filter applying hand-crafted rules based on

---

[8]`http://edits.fbk.eu/`

NEs to detect cases where no entailment was found. Despite the simplicity of the approach, applying the NE filter yielded a small improvement in precision. Iftene (2009), Iftene and Moruz (2009) and Rodrigo et al. (2009) did both dependency parsing and NE recognition. In particular, Iftene and Moruz (2009)'s ablation test showed that the NE module contributed 11.55%, 5.2%, and 6.17% accuracy on RTE-3, RTE-4, and RTE-5 datasets respectively, substantially more than other resources. Our work introduced in Chapter 4 can also be viewed as NE-centered RTE modules.

Predicate-argument structure in the Propbank style has been proved to be useful for many NLP tasks. Both Glinos (2009) and Bensley and Hickl (2009) used semantic parsers to obtain semantic dependencies between words. Nielsen et al. (2009) proposed a facet-based representation, which is more abstract than the normal dependency structure, and different from the Propbank semantic dependencies. Another variant (Ofoghi and Yearwood, 2009) utilized a Link Grammar Parser (Sleator and Temperley, 1993) to extract propositions and check entailment on top of two propositions. In the work described in Chapter 8, We use the Propbank-styled semantic dependencies and treat them as important information for relatedness and entailment recognition.

Instead of using one representation, Mehdad et al. (2009a) jointly represented syntactic and semantic dependencies using a Syntactic Semantic Tree Kernel. Sammons et al. (2009) proposed a Modular Representation and Comparison Scheme (MRCS) to incorporate multiple levels of annotations on the same text. Each resource generated a separate view of the underlying text, or augmented a view produced by another tool (specifically, modality and quantifiers augmented the views generated by semantic role labelers). This idea is consistent with our unified representation based on dependency relations presented in Section 9.1.

There is another interesting work in terms of representing meaning of text. Siblini and Kosseim (2009)'s system automatically acquired an ontology representing the text fragment and another one representing the hypothesis, and then aligned the created ontologies to determine the entailment relation.

In the end, we discuss the canonical representation for inference, that is the logic form. Clark and Harrison (2009a,b) developed a system called BLUE (Boeing Language Understanding Engine), which firstly created a logic-based representation of a text **T** and then performed simple inference to try and infer a hypothesis **H**. The Monte Carlo Pseudo Inference

Engine for Text (MCPIET) (Bergmair, 2009) addressed the RTE problem within a new theoretic framework for robust inference and logical pattern processing based on integrated deep and shallow semantics. They pointed out the tradeoff between *informativity* and *robustness*. They proposed an important new notion of the degree of validity, and provided some evidence to suggest that this concept played a crucial role in the robustness of shallow inference. At the same time, their framework still supported informationally rich semantic representations and background theories, which played the central role in the informativity of deep inference. The solution was called Monte Carlo Semantics.

2.3 Entailment Recognition

After the meaning representation derivation, the relation between the representations needs to be discovered. In this section, we focus on several aspects of the approaches to entailment recognition after preprocessing. Notice that these approaches are not mutually exclusive.

2.3.1 Logic Inference

Probably the most straightforward way is to consider the logic inference. Riabinin (2008) presented an overview of RTE systems using logical inference, starting from RTE-1, with an emphasis on how those systems overcame the need for a large amount of background knowledge. In this subsection, we only pick up some representative systems. We assume the logic form for the text has been acquired, and thus, only focus on the "inference" part.

Tatu et al. (2006) proposed a knowledge representation model and a logic proving setting with axioms on demand. They developed two slightly different logical systems with the third lexical inference system, which boosted the performance of the deep semantic oriented approach on the RTE data

One of the first efforts to combine shallow NLP methods with a deep semantic analysis was made by Bos and Markert (2005). Then, Bos and Markert (2006) combined two approaches, a shallow method based mainly on word-overlap and a method based on logical inference, using first-order theorem proving and model building techniques. They used a machine learning technique to combine features from both methods.

MacCartney and Manning (2007) presented a computational model of natural logic for textual inference. They aimed at a middle way of robust systems sacrificing semantic precision and precise but brittle systems relying on first-order logic and theorem proving. Their system found a low-cost edit sequence which transformed the premise (i.e., the text) into the hypothesis; learned to classify entailment relations across atomic edits; and composed atomic entailments into a top-level entailment judgment. They evaluated their model mainly on the FraCaS test suite.

The BLUE system (Clark and Harrison, 2009a,b) created a logic representation of **T** and then performed simple inference to try and infer **H**. Ablation studies suggested that WordNet substantially improved the accuracy scores, while parsing and DIRT only marginally improved the accuracy scores. They summarized the primary challenges for their system in terms of noise in the knowledge sources, lack of world knowledge, and the difficulty of accurate syntactic and semantic analysis, which are the challenges for the whole RTE community as well.

2.3.2   Textual Rule Application

As several available inference rule collections have already been introduced (Section 2.1.3), we focus on applying these rules to the RTE task in this section.

In the approach described by Clark et al. (2007), semantic parsing to clause representation was performed and positive entailment was claimed only if every clause in the semantic representation of **T** semantically matched some clause in **H**. The only variation allowed consisted of rewritings derived from WordNet and DIRT. Given the preliminary stage of their system, the overall results showed very low improvement over a random classification baseline.

Bar-Haim et al. (2007) implemented a proof system using rules for generic linguistic structures, lexical-based rules, and lexical-syntactic rules (these obtained with a DIRT-like algorithm on the first CD of the Reuters RCV1 corpus). The entailment considered not only the strict notion of proof but also an approximate one. Given **T** and **H**, the lexical-syntactic component marked all lexical noun alignments. For every pair of alignment, the paths between the two nouns were extracted, and the DIRT algorithm was applied to obtain a similarity score. If the score was above a threshold, the rule was applied. However, these lexical-syntactic rules

were only used in about 3% of the attempted proofs and in most cases there was no lexical variation.

Iftene and Dobrescu (2007) used DIRT in a more relaxed manner. A DIRT rule was employed in the system if at least one of the anchors matched in **T** and **H**, i.e., they used the DIRT rules as unary rules. However, the detailed analysis of the system that they provided showed that the DIRT component was the least relevant one (adding 0.4% of precision).

In (Marsi et al., 2007), the focus was on the usefulness of DIRT. In their system, a paraphrase substitution step was added on top of a model based on a tree alignment algorithm. The basic paraphrase substitution method followed three steps. Initially, the two patterns of a rule were matched in **T** and **H** (instantiations of the anchors X and Y did not have to match). The text tree was transformed by applying the paraphrase substitution. Following that, the transformed text and hypothesis trees were aligned. The coverage (proportion of aligned content words) was computed and if it was above a certain threshold, entailment was true. The paraphrase component added 1.0% to development set results and only 0.5% to test sets, but a more detailed analysis on the results of the interaction with the other system components was not given.

Nevertheless, DIRT is still one of the largest available collection of its kind and it has a relatively good accuracy (in the 50% range for top generated paraphrases, (Szpektor et al., 2007)). In Chapter 5, we present our work on applying DIRT to RTE after an extension and refinement of the rule collection.

Szpektor and Dagan (2008) acquired unary rules instead of the binary DIRT-style rules and showed improvement on the accuracy, although it is still far from satisfactory. In order to make the rule application more precise, Basili et al. (2007) and Szpektor et al. (2008) proposed attaching selectional preferences to inference rules. Those are semantic classes which correspond to the anchor values of an inference rule and have the role of making the precise context in which the rule can be applied[9]. However, in this dissertation we investigate the first and more basic issue: how to successfully use rules in their original form (Chapter 5).

---

[9]For example, *X won Y* entails *X played Y* only when Y refers to some sort of competition, but not if Y refers to a musical instrument.

2.3.3   Similarity Measurements

Based on different meaning representations (Section 2.2), various similarity functions are defined. Besides the common cosine similarity, Levenshtein distance, and so on (Galanis and Malakasiotis, 2009, Malakasiotis, 2009), Agichtein et al. (2009) incorporated several substring similarity scores, and both Castillo and i Alemany (2009) and Pakray et al. (2009) applied longest common substring (LCS) algorithms.

As being mentioned before (Section 2.1.2), WordNet, VerbOcean, and Wikipedia were used to calculate similarity scores between two words or entities. They provided alternatives to the string matching algorithms, and added semantics in.

On top of the similarity between words or entities, Krestel et al. (2009a,b) defined a similarity function between two syntactic predicate-argument structures, and Yatbaz (2009) defined a similarity score between dependency paths. The main module described in (Iftene, 2009) and (Iftene and Moruz, 2009) accepted outputs from all the other modules and used a global similarity score (they named it the fitness value) to decide on the final answer.

One problem with the similarity measurement is that the directionality of the entailment relation is ignored. Most of the similarity functions were defined as a symmetric relation. In Chapter 8, it is shown that instead of recognizing textual entailment, similarity-based approaches are more suitable for recognizing non-directional relations, e.g., the relatedness.

One category of asymmetric similarity measurement is the edit distance models. Both Cabrio et al. (2009) and Mehdad et al. (2009b) developed their system based on the EDITS package, which computed the **T**-**H** distance as the cost of the edit operations (i.e., *insertion*, *deletion*, and *substitution*) that were necessary to transform **T** into **H**. The directionality ensured the non-symmetric of this distance-based measurement.

2.3.4   Matching and Alignment

Almost every RTE system contains a module to match or align some parts of **T** and **H** at some stage of the whole procedure. In some approaches, the matching or alignment module can directly output the final answer; while in other methods, the results are passed to the later stages for

further processing.

In the first category, Li et al. (2009b) designed different strategies to recognize true entailment and false entailment. While the similarity between **T** and **H** was measured in order to recognize true entailment, the exact entity and relation mismatch was used to recognize the false entailment. Ofoghi and Yearwood (2009) decided whether the entailment held between two propositions based on the assumption that each single proposition in **H** needs to be entailed at least by the meaning of one proposition in **T**. Glinos (2009) determined the entailment relation based on matching two predicate-argument structures.

In the second category, Padó et al. (2009b) used a phrase-based aligner, MANLI (MacCartney et al., 2008), and took the output as features for entailment recognition. Yatbaz (2009) and Siblini and Kosseim (2009) presented similar approaches. The difference was that the former performed the alignment between two dependency trees, while the latter aligned two ontologies. Sammons et al. (2009) clarified the two distinct alignment models and argued that the goal of an ideal alignment was to make local decisions, instead of being a global scoring function for the entailment decision.

2.3.5   Feature-based Classification

Both the logic-rule-based and textual-rule-based systems suffer from either a laborious and fragile module with hand-crafted rules (i.e., lack of recall) or a large collection of "noisy" rules (i.e., lack of precision). In order to avoid these disadvantages, people usually treat RTE as a classification task and apply feature-based machine learning techniques to obtain the answer.

For example, Agichtein et al. (2009), Balahur et al. (2009), and Ferrández et al. (2009) took string similarity scores as features; Rodrigo et al. (2009) had features from both dependency parsing and NE recognition; Nielsen et al. (2009) extracted features from the facet-based representations; and Bensley and Hickl (2009) extracted features from the predicate-argument structures.

An alternative to the feature engineering attempts, support vector machines (SVMs) with different kernels are also popular in this classification task. Both the (constituent) tree kernel (Zanzotto et al., 2009, Mehdad et al., 2009a) and the subsequence kernel based on syntactic dependency

paths (Wang and Neumann, 2007a) were quite successful.

In this dissertation, we also follow the classification approaches and mostly use the same SVM-based classifier for consistency. However, instead of using the tree kernels, we explicitly extract features based on both syntactic and semantic dependency paths (or triples) as an approximation of the meaning, which greatly reduce the number of dimensions of the feature vectors and make the (intermediate) results more explainable.

2.4   Related Tasks

RTE has a close relationship to several other tasks, which also deal with relations between pairs of text. Contradiction recognition is a natural extension of the traditional two-way RTE task; paraphrase acquisition has been widely studied, which can be viewed as a bi-directional entailment; and the key feature of entailment, directionality, has not been fully explored yet.

Notice that the tasks introduced in this section are different from downstream applications of RTE, such as summarization, information extraction, and so on. Instead, they are part of or in parallel to entailment recognition. The applications of RTE are discussed in the next section.

2.4.1   Contradiction Recognition

An extension to the traditional two-way RTE task has been proposed in the RTE-3 pilot task. While preserving ENTAILMENT, they divide non-entailment cases into two sub-classes, CONTRADICTION and UNKNOWN. Contradiction was rare in the RTE-3 test set, occurring in only about 10% of the cases, and systems found accurately detecting it difficult (Voorhees, 2008).

de Marneffe et al. (2008) treated detecting conflicting statements as a foundational text understanding task. They proposed a definition of contradiction for NLP tasks and developed available corpora, from which they constructed a typology of contradictions. Detecting some types of contradiction required deeper inferential paths than their system was capable of, but they achieved good performance on types arising from negation and antonymy.

Murakami et al. (2009) focused on *agreement* and *conflict* recognition from subjective texts, i.e., opinions. They discussed how to efficiently collect valid examples from Web documents by splitting complex sentences into fundamental units of meaning called *statements* and annotating relations at the statement level. The *conflict* cases contained three finergrained categories: *contradiction*, *confinement*, and *conflicting opinion.*

2.4.2 Paraphrase Acquisition

Paraphrase can be viewed as a bi-directional entailment relation. There is a rich literature on this research topic, e.g., Shinyama et al. (2002), Barzilay and Lee (2003), and so on. We cannot cover all the aspects of paraphrase acquisition and application in this subsection, but only those related to RTE. Androutsopoulos and Malakasiotis (2010) did a survey on common approaches to paraphrasing and entailment recognition.

Paraphrase acquisition is mostly done at the sentence-level (Barzilay and McKeown, 2001, Dolan et al., 2004), which cannot be directly used as a resource for other NLP applications. At the sub-sentential level, interchangeable patterns (Shinyama and Sekine, 2003) are extracted, which are quite successful in named-entity-centered tasks, like information extraction, while they are not generalized enough to be applied to other tasks.

In machine translation, translation phrase pairs can be extracted from bilingual parallel or comparable corpora (Fung and Lo, 1998, Vogel, 2003, Wu and Fung, 2005). Munteanu and Marcu (2006) extracted sub-sentential translation pairs from comparable corpora based on the log-likelihood-ratio of word translation probability. They exploited the possibility of making use of reports within a limited time window, which were about the same event or having overlapping contents but in different languages. Quirk et al. (2007) extracted fragments using a generative model of noisy translations. They showed that even in non-parallel corpora, useful parallel words or phrases can still be found and the size of such data is much larger than that of parallel corpora. Therefore, in a similar manner, sub-sentential paraphrase fragment pairs can also be extracted from monolingual comparable corpora. We present a pilot study on this issue in Section 5.6.

Instead of being used as a resource for RTE, paraphrase acquisition can be tackled in parallel to entailment recognition. A recent work by

Heilman and Smith (2010) proposed a generic system based on a tree editing model to recognize textual entailment, paraphrase, and answers to questions in one unified framework. The model was used to represent sequences of tree transformations involving complex reordering phenomena and shown to be a simple, intuitive, and effective method for modeling pairs of semantically related sentences. They described a logistic regression model that used 33 syntactic features of edit sequences to classify the sentence pairs. In Chapter 9, we also propose a model of recognizing different semantic relations between pairs of text simultaneously and we compare the results with Heilman and Smith (2010) as well.

2.4.3   Directionality Recognition

In contrast to the paraphrase acquisition, there is little work on the directionality recognition. The DIRT algorithm (Lin and Pantel, 2001) does not guarantee to extract directional inference rules, since the similarity measurement is symmetric (Lin, 1998). However, (true) entailment is not bi-directional.

Chklovski and Pantel (2004) extracted specific directional relations between verbs, but did not generalize the approach for other relations. Bhagat et al. (2007) defined the directionality between two named-entity relations based on distributional hypothesis (Harris, 1954). The intuition is that the more frequently one relation occurs, the more likely it is more general; otherwise, it is more specific. However, the distributional hypothesis does not exclude relations with strong negative polarity, like antonyms. Kotlerman et al. (2009) investigated the nature of directional similarity measures between lexicons, which aimed to quantify distributional feature inclusion. They identified desired properties of such measures, specified a particular one based on averaged precision, and demonstrated the empirical benefit of directional measures for lexical expansion.

Another line of research was done by Danescu-Niculescu-Mizil et al. (2009) and Danescu-Niculescu-Mizil and Lee (2010). They presented an algorithm to learn linguistic constructions that, like "doubt", which is downward entailing. Their algorithm was unsupervised, resource-lean, and effective, accurately recovering many downward entailing operators that were missing from the hand-constructed lists that RTE systems currently used. Furthermore, they also proposed an approach that could

be applied to many languages for which there is no pre-existing high-precision database of negative polarity items.

All this work has been done at the lexical level, which is different from the granularity considered in this dissertation. To our best knowledge, in the context of RTE, there is no separation between directional entailment and paraphrase. In Chapter 6, we elaborate on this issue.

2.5 Performance of the Existing Systems

The main evaluation metric for the RTE systems is *accuracy*, i.e., the percentage of matching system judgments compared against the gold standard compiled by the human assessors. Currently, other measurements like efficiency are not the focus of the community.

Based on the intuition that entailment is related to the similarity between text and hypothesis, Mehdad and Magnini (2009) provide several RTE baselines on top of the BoW representation and different similarity estimated as the degree of word overlap between **T** and **H**. On the RTE-3 dataset, different settings vary from 0.585 to 0.625; while on the RTE-4 dataset, the results vary from 0.510 to 0.587. Both are on the two-way annotated data, ENTAILMENT vs. non-entailment.

As for the system performance in the yearly RTE challenges, the average accuracy of the participating systems is around 60% on the two-way annotated data and with a 5-10% drop on the three-way annotated data (ENTAILMENT, CONTRADICTION, and UNKNOWN). The full results can be found in the overview papers of the challenges (Giampiccolo et al., 2007, 2009, Bentivogli et al., 2009). The results of the top five participating systems[10] are listed as follows, with two-way annotation (Table 2.2) and three-way annotation (Table 2.3) respectively (our results are shown in bold).

2.6 Applications

One of the original motivations for RTE is to provide a generic semantic engine, which serves for other NLP tasks. In practice, RTE systems have been widely used as components for other systems as well.

---

[10]We use the first author's last name as the indicator for their participating system and we keep the old indicator even if their author list changed later.

Rank	RTE-3		RTE-4		RTE-5	
	System	Accuracy	System	Accuracy	System	Accuracy
1	Hickl	0.800	Hickl	0.746	Iftene	0.735
2	Tatu	0.723	Iftene	0.721	**Wang**	**0.685**
3	Iftene	0.691	**Wang**	**0.706**	Li	0.670
4	Adams	0.670	Li	0.659	Mehdad	0.662
5	**Wang**	**0.669**	Balahur	0.608	Sammons	0.643

Table 2.2: Top five participating systems in the RTE challenges (two-way annotation)

Rank	RTE-4		RTE-5	
	System	Accuracy	System	Accuracy
1	Iftene	0.685	Iftene	0.683
2	Siblini	0.616	**Wang**	**0.637**
3	**Wang**	**0.614**	Ferrández	0.600
4	Li	0.588	Malakasiotis	0.575
5	Mohammad	0.556	Breck	0.570

Table 2.3: Top five participating systems in the RTE challenges (three-way annotation)

Harabagiu and Hickl (2006) demonstrated how RTE systems can be used to enhance the accuracy of current open-domain question answering systems. In their experiments, they showed that when textual entailment information was used to either filter or rank answers returned by a QA system, accuracy would be increased by as much as 20% overall. Celikyilmaz and Thint (2008) used an RTE module to rank the retrieved passages/sentences by matching the semantic information contained in the retrieved sentences and the given questions. Sentences with a high rank are likely to contain the answer phrases.

Roth et al. (2009) defined the problem of recognizing entailed relations - given an open set of relations, find all occurrences of the relations of interest in a given document set - and posed it as a challenge to scalable information extraction and retrieval. They argued that textual entailment was necessary to solve the common problems: supervised methods were not easily scaled, while unsupervised and semi-supervised methods were restricted to frequent, explicit, highly localized patterns. They implemented a solution showing that an RTE system can be scaled to a much larger information extraction problem than that represented by the RTE challenges.

Mirkin et al. (2009b) addressed the task of handling unknown terms in statistical machine translation. They proposed using source-language monolingual models and resources to paraphrase the source text prior to translation. They allowed translations of entailed texts rather than paraphrases only. Their experiments showed that the proposed approach substantially increased the number of properly translated texts. Instead of improving the MT systems, Padó et al. (2009a) proposed a metric that evaluated MT output based on a rich set of features motivated by textual entailment, such as lexical-semantic (in-)compatibility and argument structure overlap. They compared that metric against a combination metric of four state-of-the-art scores in two different settings. The combination metric outperformed the individual scores, but was beated by the entailment-based metric.

Many participating systems in Answer Validation Exercise (AVE) at the Cross Language Evaluation Forum (CLEF) (Peñas et al., 2007, Rodrigo et al., 2008) utilized RTE systems as core engines. The AVE task asked the participating systems to validate answers output by the QA systems, and it can be easily transformed into an RTE problem by combining question and answer pairs into **H**s and taking documents as **T**s. We also participated in the exercises and achieved the best result for English (Wang and Neumann, 2007b) and for German (Wang and Neumann, 2008a).

2.7 Summary

In this chapter, the related work in the field is reviewed. We start with data resources and knowledge resources, including the available annotated datasets and textual inference rule collections. Then two important procedures followed by most of the RTE systems, meaning representation derivation and entailment relation recognition, are described. We go through a number of RTE approaches proposed in the recent years and classify them into different categories, and also introduce several related tasks, contradiction recognition, paraphrase acquisition, and directionality recognition. Finally, we present the state-of-the-art RTE system performance and several successful downstream applications.

In the rest of this dissertation, we describe our approaches to RTE, as well as other related tasks.

Part A: Intrinsic Approaches

3 An Extensible Architecture for RTE

In this chapter, we introduce the architecture of our (intrinsic) approaches to the RTE task. We firstly discuss the complexity of the problem as well as the motivation of our approach. We then show the difference between the common RTE system architecture and our proposal. Since the details of the approach are elaborated on in the following two chapters, this chapter only gives an overview. Finally, we introduce some possible extensions of the current approach together with some related work done by other researchers.

3.1 Motivation of the Approaches

Let us take a look at the following example:

T: <u>Bush$_1$</u> *used his weekly radio address to try to build support for <u>his$_6$</u> <u>plan$_2$</u> <u>to allow workers to$_3$</u> divert part of$_7$ their <u>Social Security payroll</u> <u>taxes$_4$</u> into private <u>investment accounts$_5$</u>.*

H: <u>Mr. Bush$_1$</u> *is <u>proposing$_2$</u> that <u>workers be allowed to$_3$</u> <u>divert$_7$</u> their <u>payroll</u> <u>taxes$_4$</u> into private <u>accounts$_5$</u>.*

This is a positive example taken from the RTE-2 corpus, which means **T** entails **H**. In order to get the final answer, we need to process a lot of information:

1. "Mr. Bush" in **H** is referring to "Bush" in **T**.

2. "Proposing" something means there is a "plan".

3. "To allow workers to" is the same meaning as "workers be allowed to", and the only difference is the syntactic variation of active and passive voices.

4. "Payroll taxes" in **H** are the same taxes as "Social Security payroll taxes" in **T**, because according to the context, that is the only possibility.

5. "Investment accounts" is a kind of "account".

6. "His" in **T** is referring to "Bush" (and "Mr. Bush" in **H**). Therefore, we can find the same connection between "Bush" and the "plan" as well as "Mr. Bush" and "proposing".

7. We also need to know "divert part of" something entails "divert" that thing, since it is monotonic. This is not the case for verbs like "deny".

If we take a closer look at these points, they range from syntactic variation to lexical semantic relations, from gerund to anaphora or coreference resolution. After using all this knowledge, we can then say **T** entails **H**.

Systematic manual analyses of RTE corpora have looked quantitatively at some of these points. For instance, for the RTE-1 dataset, Vanderwende et al. (2006) showed that 37% of the data could be solved merely at the syntactic level, and if a general-purpose thesaurus (e.g., WordNet) was additionally exploited, that number increased to 49%.

A more detailed study has been done for the RTE-2 dataset. Garoufi (2007) manually annotated 400 positive examples as well as some negative ones. She used an inventory of 23 linguistic features, including *acronym*, *hypernym*, *apposition*, *passivization*, *nominal*, *modifier*, and so on. 22 of these features were observed in the data, ranging from *negation* (2 pairs) to *identity* (365 pairs). These features were further grouped into five categories, identity, lexicon, syntax, discourse, and reasoning. Each **T-H** pair in the dataset contains one or more categories of features.

For the RTE-3 dataset, Clark et al. (2007) manually annotated 100 positive cases with additional information. Besides the linguistic analysis, they also discovered that 18 **T-H** pairs require general world knowledge (i.e., common facts about the world), 7 pairs require core theories (i.e., space and time), and 11 pairs require knowledge related to frames and scripts (i.e., stereotypical situations and events).

Based on these previous analyses, it is obvious that RTE is a challenging task due to the rich linguistic phenomena and high knowledge requirement. Empirical results also confirm this, with an average system performance of about 60% accuracy (Section 2.5). Issues affecting performance include not only the wide variety of linguistic analysis required, but also limited training data (for machine learning based systems) and error propagation due to long pipelines. Furthermore, the knowledge that is required to determine the entailment could be beyond the text of the **T-H** pair.

Consequently, we propose a novel extensible architecture which consists of a number of specialized RTE modules. Each module deals with a subset of the corpus, ideally targeting one linguistic phenomenon. Since it does not need to cover the whole dataset, the requirement for the size of the training data becomes less severe. Instead, we separate the dataset into subsets and train those modules individually. The idea is quite similar to the famous *divide-and-conquer* algorithm (Knuth, 1998).

Now, the main issues are:

- What is a good subset of the data?

- What kind of specialized modules should be designed?

These are the main topics of the next section.
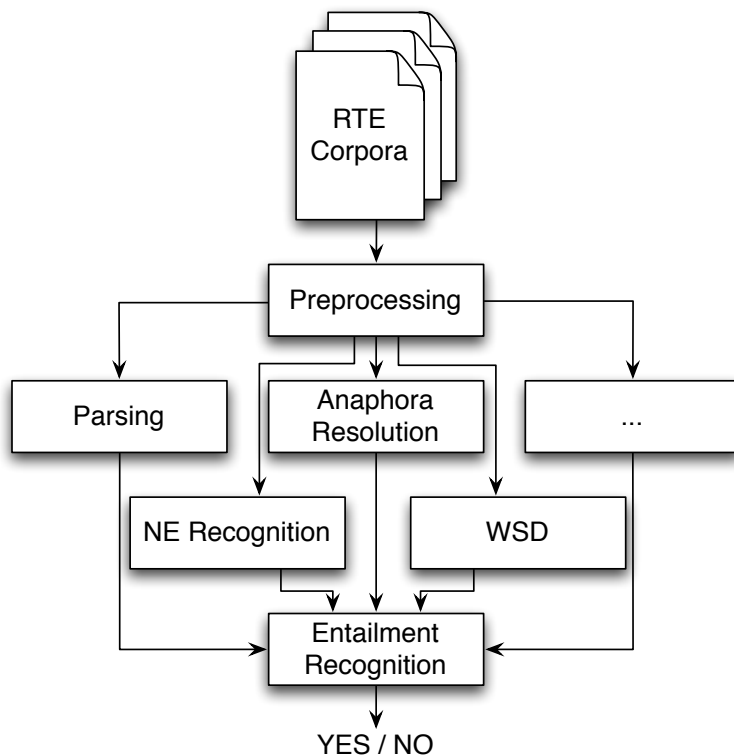
Figure 3.1: The traditional RTE system architecture

3.2   The Architecture

Figure 3.1 shows the common architecture of many machine-learning-based (ML-based) RTE systems. Basically, it contains three steps, pre-processing, linguistic processing and feature extraction, and post-processing (ML-based classification). This works well when we have simpler tasks (less features) and large amounts of annotated data. However, this is not the case for RTE.

Alternatively, we propose another architecture, shown in Figure 3.2. Instead of processing the whole dataset using one integrated system, we split the corpus into subsets and tackle them with different subsystems, i.e., specialized RTE modules. We then need a good splitter to separate different cases of entailment[1] and appropriate modules to handle them separately.

There is a typology of the linguistic phenomenon of implication in the literature by Chierchia and McConnell-Ginet (2000). It contains *(strict) entailment*, *conventional implicature*, *conversational implicature*,

---

[1]As we mentioned before in the Chapter 1, the term "entailment" used in this dissertation is in fact "implication" in the traditional linguistic literature.
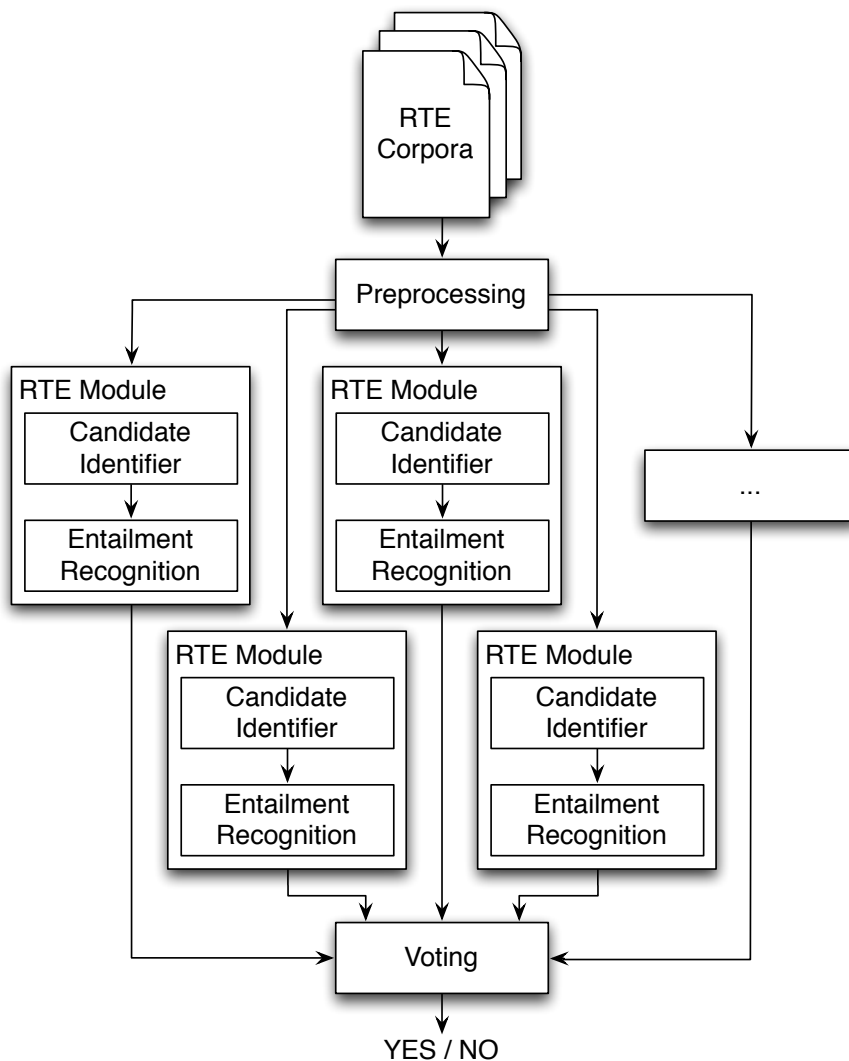
Figure 3.2: The proposed RTE system architecture

paraphrase, and so on. However, these cases are not trivially machine-differentiable. Nor are they suitable for the data collected from different NLP applications.

In fact, the criteria for a good subset are highly related to the module dealing with it. Therefore, it is easier to do the split based on what specialized modules we have. For instance, if we have an inferencer dealing with temporal expressions, we should find those cases of entailment contained in the dataset which need temporal reasoning. If we have an accurate person name normalization system, we should find those cases that need pronoun resolution. In the more general sense, we need to discover those **T-H** pairs which the available systems can handle well.

Therefore, we prefer a system with *high precision* over one with high

recall (if we cannot achieve them both at the same time) in both the splitting of the data and the processing with the specialize modules. In particular, the criteria for such an architecture are:

A good split using basic linguistic processing to choose a subset of the whole dataset;

A good module precision-oriented, preferring accuracy to coverage of the dataset.

In the following, we briefly introduce our RTE system based on this architecture (Figure 3.2) and the details are in Chapter 4.

For preprocessing, we utilize several linguistic processing components, such as a POS tagger, a dependency parser, and a named-entity (NE) recognizer to annotate the original plain texts from the RTE corpus. We then apply several specialized RTE modules. Since all the modules aim at high precision, they do not necessarily cover all the **T-H** pairs. The cases which cannot be covered by any specialized RTE module are passed to the high-coverage, but probably less accurate backup modules. In the final stage, we join the results of all specialized RTE modules and backup modules together. Different confidence values are assigned to the different modules according to the performances on the data for development. In order to deal with possible overlapping cases (i.e., **T-H** pairs that are covered by more than one module), a voting mechanism is applied taking into account the confidence values.

For the specialized modules, we have developed and implemented the following three to deal with three different cases of entailment:

Temporal anchored pairs Extract temporal expressions and corresponding events from the dependency trees, and apply entailment rules between extracted time event pairs;

Named entity pairs Extract other Named Entities (NE) and corresponding events, and apply entailment rules between extracted entity-event pairs;

Noun phrase anchored pairs For pairs with no NEs but containing two NPs, determine the subtree alignment, and apply a kernel-based classifier.

In addition to the precision-oriented RTE-modules, we also consider two robust but not necessarily precise backup strategies to deal with those

cases which cannot be covered by any specialized module. Chief requirements for the backup strategy are robustness and simplicity. Therefore, we considered two backup modules, the Triple backup and the Bag-of-Words (BoW) backup (Wang and Neumann, 2007a).

The Triple backup module is based on the Triple similarity function which operates on two triple (dependency structure represented in the form of *<head, relation, modifier>*) sets and determines how many triples of **H** are contained in **T**. The core assumption here is that the higher the number of matching triple elements, the more similar both sets are, and the more likely it is that **T** entails **H**. The function uses an approximate matching function. Different cases (i.e., ignoring either the parent node or the child node, or the relation between nodes) may provide different indications for the similarity of **T** and **H**. We then sum them up using different weights and divide the result by the cardinality of **H** for normalization.

The BoW backup module is based on BoW similarity score, which is calculated by dividing the number of overlapping words between **T** and **H** by the total number of words in **H** after a simple tokenization according to the space between words.

There is one more issue we have not addressed, which is the application of the external knowledge. Chapter 5 focuses mainly on this. In particular, we consider using a collection of textual inference rules for the RTE task. The rules were obtained separately, using an acquisition method based on the Distributional Hypothesis (Harris, 1954). The system itself can be viewed as an extended version of the third specialized module mentioned above. The original module extracts *Tree Skeleton* (Section 5.4.2) from the dependency trees and applies a subsequence-kernel-based classifier that learns to decide whether the entailment relation holds between two texts. The extended system replaces the learning part with the rule application. Therefore, whether the inference rule triggers defines the subset of the data which the specialized module deals with.

3.3 Summary

In summary, this chapter provides an overview of the extensible architecture of our RTE system. The system contains multiple specialized modules which deal with different types of entailment separately, instead of tackling them all together. We show three such modules in Chapter 4

and one extended module with an external inference rule collection in Chapter 5.

Bobrow et al. (2007) also had the idea of developing a precision-oriented RTE system, although their system had very limited coverage of the dataset. Bentivogli et al. (2010) built specialized datasets made of monothematic **T-H** pairs, i.e., pairs in which a certain phenomenon relevant to the entailment relation is highlighted and isolated. Recent work done by Mirkin et al. (2010a) focused on those data with discourse information involved. All this related work confirms the "specialized" strategy of tackling the RTE task.

Naturally, more specialized modules (including those mentioned above) can be added into our extensible architecture. For example, one can enhance entailment recognition with logic inferencing, which deals with quantifiers, modal verbs, etc. The integration of generic and specialized modules is also outside the scope of this dissertation. In the long run, we will explore different combination strategies as well. We leave these issues for the future work (Chapter 10).

4 Textual Entailment with Event Tuples

In this chapter[1], we firstly introduce one specialized module for tackling textual entailment pairs with temporal expressions. A separate Time Anchoring Component (TAC) is developed to recognize and normalize the temporal expressions contained in the texts. The corresponding events can then be extracted from the dependency trees. We define time-event pairs to partially represent the meaning of the sentences, and on top of that, the entailment relation can be verified via simple rules. In addition, we show the generalization of this module can handle texts containing other types of named-entities as well, i.e., locations, persons, and organizations. The final event tuple contains time, location, and a list of participants (either persons or organizations). The experiment results show the advantage of such precision-oriented specialized RTE modules and suggest a further integration into a larger framework for general textual inference systems.

---

[1]Section 4.1 to Section 4.5 have been published in (Wang and Zhang, 2008), and it was a collaboration with Yajing Zhang, who focused on developing the module handling temporal expressions. Section 4.7 has been published in (Wang and Neumann, 2009), and it was a collaboration with PD Dr. Günter Neumann.
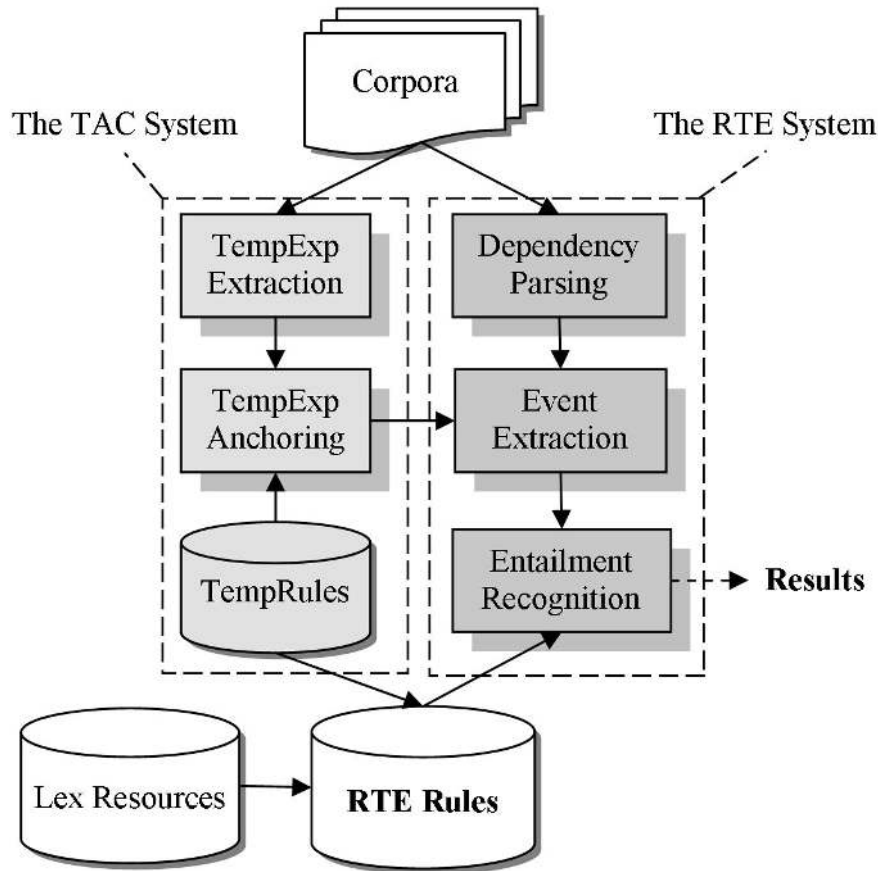
Figure 4.1: Architecture of the TACTE System.

4.1   System Architecture

Figure 4.1 shows the basic architecture of our TACTE system[2]. The system mainly consists of two components, RTE and TAC, and the entailment rules serve as the knowledge base to detect the entailment relation. The TAC system uses SProUT, an information extraction (IE) platform (Section 4.2), to extract *Date* and *Time* expressions and anchor them based on manually defined rules. The RTE system pre-processes the texts using a dependency parser and later extracts the corresponding events based on the dependency structure using the temporal expressions as starting points (Section 4.3). The entailment rules (Section 4.4) come from two sources: 1) lexical semantic resources and 2) entailment rules between temporal expressions. Notice that extra components can be added into this extensible architecture, which will be described in Section 4.7.1. In the following sections, we illustrate these components in

---

[2]TACTE stands for *Time Anchoring Component for Textual Entailment*.

detail.

4.2 Temporal Expression Anchoring

The use of temporal expressions is based on the assumption that very often important clues to distinguish what belongs to the main topic of a text and hypothesis and what is subsidiary information are given by temporal information. However, temporal information about the temporal location of events is not always given explicitly by some date or time expression, but by relative references such as "the week before". Therefore, a Time Anchoring Component (TAC) is developed to resolve temporal expressions, construct a time line of events, and distinguish event information from other information.

The core engine extracting temporal expressions in TAC is provided by SProUT[3] (Drozdzynski et al., 2004), a multilingual platform developed for shallow natural language processing applications. SProUT combines finite state techniques with unification of typed feature structures (TFSes). TFSes provide a powerful device for representing and propagating information. Rules are expressed by regular expressions over input TFSes that get instantiated by the analysis. The uniform use of TFSes for input and output also allows for cascaded application of rule systems.

The representation of dates and times in TAC is based on *OWLTime* (Hobbs and Pan, 2006). This ontology provides classes for representing temporal instants and durations. The core date-time representation is the class *DateTimeDescription* that provides as properties fields for representing the day, month, year, hour, minute, second, weekday as well as the time zone. The use of *OWLTime* presupposes to some extent that dates or times are completely specified. But it poses some problems for the representation of partial and underspecified temporal expressions as used in natural language texts. The TAC component described here bridges the gap between temporal natural language expressions and *OWLTime* representations.

Both time points and durations are represented by the class *DateTimeInterval*[4] which references *DateTimeDescription* and *DurationDe-*

---

[3]`http://sprout.dfki.de/`

[4]A time point described by a *DateTimeDescription* can be viewed as an interval according to its granularity or specificity, e.g., "yesterday" is an interval of the last 24 hours preceding the last midnight.

scription. For better compliance with *TimeML*[5], *OWLTime* was extended by adding to the *DateTimeDescription* class properties for representing the week number (e.g., for representing the reference of expressions like "last week"), seasons (e.g., for references of "last summer") and daytimes (e.g., "afternoon") rather than representing these imprecise times directly as durations.

4.2.1   Two Types of Temporal Expression

In the temporal expression extraction process we distinguish two types of temporal expressions: *time points* and *durations*.

Time Points   *DateTimeInterval* only specifies *DateTimeDescription* with following properties: *day, month, year, hour, minute, second, part-of-day (pofd), day-of-week (dofw), weeknumber, part-of-month (pofm), part-of-year (pofy)*. Among all these features, the feature *year* is obligatory which means each anchored time point must at least specify a value for *year*. Figure 4.2 (top) shows the representation for the date "Friday October 24th, 1997", omitting the namespace prefixes for presentation purposes.

An important dimension to take into account for temporal resolution and computation is the *granularity* order of these features. The order is similar to our intuition:

$[second < minute < hour < pofd < dofw < day < weeknumber < pofm < month < pofy < year]$

Durations   *DateTimeInterval* can consist of *DateTimeDescription* or *DurationDescription*. *DurationDescription* contains properties of *days, months, years, hours, minutes, seconds, weeks*. Additionally thirteen relations defined in (Allen, 1983) describe the relation between *DurationDescription* and the reference time.

Due to the restricted granularity level of the reference time, a *DateTimeInterval* may have underspecified beginning and end points. Figure 4.2 (bottom) shows the representation for "from Tuesday to Thursday", where the reference time is "October 24th, 1997" (Friday).

We illustrate this using the following example, where our reference time is set to be "October 24th, 1997" (Friday) (cf. Figure 4.2).

---

[5]http://www.timeml.org/site

$$
\begin{bmatrix}
DateTimeInterval \\
\\
\text{HASDATETIMEDESCRIPTION} & \begin{bmatrix} DateTimeDescription \\ \text{DAY} & 24 \\ \text{MONTH} & 10 \\ \text{YEAR} & 1997 \\ \text{DOFW} & 5 \end{bmatrix}
\end{bmatrix}
$$

$$
\begin{bmatrix}
DateTimeInterval \\
\text{HASBEGINNING} & \begin{bmatrix} DateTimeDescription \\ \text{DOFW} & 2 \end{bmatrix} \\
\\
\text{HASEND} & \begin{bmatrix} DateTimeDescription \\ \text{DOFW} & 4 \end{bmatrix} \\
\\
\text{INTERVALBEFORE} & \#refDateInstance
\end{bmatrix}
$$
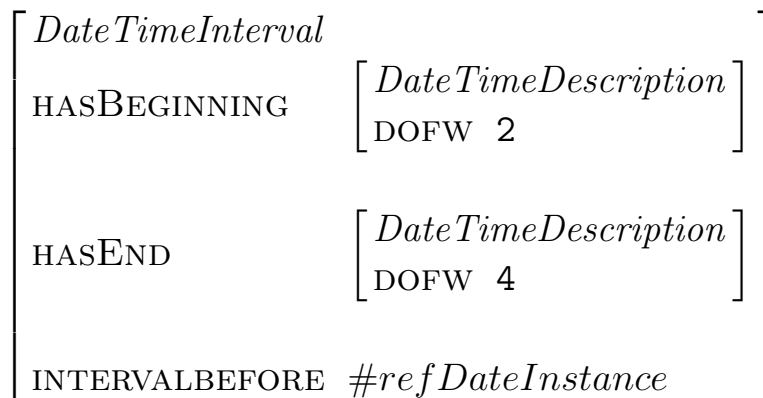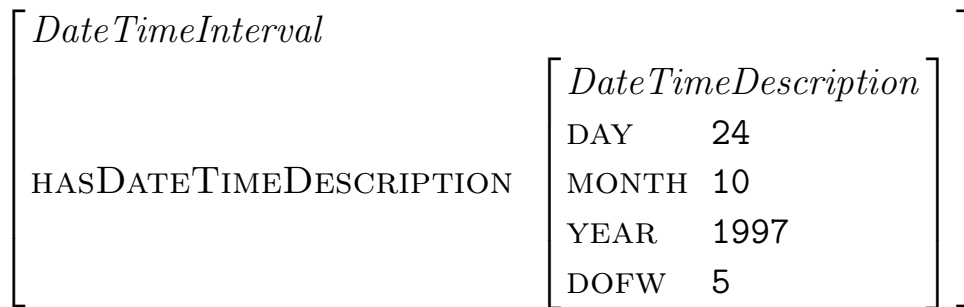
Figure 4.2: TFS of "Friday October 24th, 1997" and TFS of "from Tuesday to Thursday"

- *The president visited an Australian produce display in Knightsbridge* **from Tuesday to Thursday***.*

This example is a *DateTimeInterval* containing two *DateTimeDescriptions* for the beginning and end respectively. Since "from Tuesday to Thursday" is before Friday on the time line, it represents a INTERVALBEFORE relation with respect to the reference time[6].

4.2.2 Anchoring of Temporal Expressions

To anchor temporal expressions, *Absolute* and *Relative expressions* are distinguished:

Absolute expressions refer to a specific time point or period of time. It can be unambiguously identified in a calendar, for instance, "June

---

[6]The reference time here is a *DateTimeDescription* instance referred by *#refDateInstance*.

6th, 2006".

Relative expressions refer to a time point or period of time that can only be unambiguously identified with the help of a reference time given by context. Examples include "yesterday", "two hours later", "in summer", and so on.

Unlike Han et al. (2006), we do not distinguish deictic and relative expressions, since both of them require a contextually given reference time to anchor the expression correctly. The difference is only in the type of context. We consider a time expression for duration as either absolute or relative expressions, for instance, "from June 6th, 2006 to June 9th, 2006", "from today to tomorrow", etc.

The reference time is context-dependent and dynamic. Currently when no absolute time is mentioned in the text or hypothesis, a default reference time is set to both[7]. Moreover, when another absolute time is mentioned in subsequent sentences, it may become the new reference time for that paragraph.

TAC also decides about the granularity level at which completion is necessary. The result inherits the granularity of the original incomplete expression. For instance, let the reference date be "October 24th, 1997" (Friday). In the following examples, the granularity of the first expression "last Wednesday" is *dofw* and is anchored to "Wednesday October 15, 1997", while the second one has the granularity of *minute* and is anchored to "October 24th, 1997, 15:08".

- *The defence secretary William Cohen announced plans on **last Thursday**.*

- *The earthquake shook the province of Mindanao at **3:08 p.m this afternoon**.*

The core of the anchored date-time are shown in Figure 4.3 where their different granularities can be observed.

Evaluated on the complete Timebank corpus (Pustejovsky et al., 2003), TAC achieves an F-measure of 82.7%. An inspection of a random selection of 200 Timebank annotations revealed a high number of annotation errors (of nearly 10% were wrongly classified). Consequently, the evaluation measures only provide an approximate value.

---

[7]For the oral discourse, the speech time is considered to be the initial reference time, and for news it is often the publication or creation time of the news. This follows Reichenbach (1999)'s speech time, event time, and reference time, but in our data, mostly, we do not need to handle the first two in one text.

$$\begin{bmatrix} DateTimeDescription \\ \text{DAY} \quad 16 \\ \text{MONTH} \quad 10 \\ \text{YEAR} \quad 1997 \\ \text{DOFW} \quad 4 \end{bmatrix} \quad \begin{bmatrix} DateTimeDescription \\ \text{MINUTE} \quad 8 \\ \text{HOUR} \quad 15 \\ \text{DAY} \quad 24 \\ \text{MONTH} \quad 10 \\ \text{YEAR} \quad 1997 \end{bmatrix}$$

Figure 4.3: Representation for "last Thursday" and "3:08 p.m this afternoon".

4.3 Event Extraction

Our event extraction algorithm is based on the dependency trees, and uses temporal expressions as starting points for the search. The dependency tree is the parsing result of a sentence using *Dependency Grammar* (DG), which consists of a bag of dependency triple relations. A dependency relation (Hudson, 1984) is an asymmetric binary relation between one token (i.e., parent node or head) and another token (i.e., child node or modifier). The dependency tree is a connected structure of all the tokens of the sentence, where each parent node can have several child nodes, but each child node can only have one parent node. The main verb (or predicate) of the sentence is the root of the tree.

The use of dependency trees is motivated by the fact that it can provide more information than shallow representations, as well as the robustness and efficiency of dependency parsing in comparison with deeper processing. Compared with constituency parsing trees, dependency structures capture the relation between individual words rather than only the constituents of the sentence.

We assume that events can be expressed by either a noun (including nominalization) or a verb. The main idea of the *EventExtraction* algorithm is to locate the temporal expression in the dependency tree and then traverse the nodes in the tree either going up or going down to find the nearest verb or noun(s). The goal of this procedure is to find the corresponding nouns or verbs which the temporal expressions modify.

For instance, if we consider the following **T-H** pair,

Algorithm 1 The *EventExtraction* Algorithm

---

    function EXTRACTEVENTS(DEPSTR, TEMPEXP): N∪V
        /* DepStr: dependency structure
           TempExp: temporal expression */
        N ← ExtractNounEvent(DepStr, TempExp)
        V ← ExtractVerbEvent(DepStr, TempExp)
    end function

    function EXTRACTNOUNEVENT(DEPSTR, NODE): N
        Find node in DepStr
        if node.POS == Noun **then**
            N ← node;
        else
            For each child in node.children
                N ← ExtractNounEvent(DepStr, child)
        end if
    end function

    function EXTRACTVERBEVENT(DEPSTR, NODE): V
        Find node in DepStr
        if node.POS == Verb **then**
            V ← node;
        else
            V ← ExtractVerbEvent(DepStr, node.Parent)
        end if
    end function

T: *Released in **1995**, Tyson returned to boxing, winning the World Boxing Council title in **1996**. The same year, however, he lost to Evander Holyfield, and in a **1997** rematch bit Holyfield's ear, for which he was temporarily banned from boxing.*

H: *In **1996** Mike Tyson bit Holyfield's ear.*

After applying our algorithm, the following events are extracted:

- 1995: released (verb);

- 1996: winning (nominalization);

- 1997: rematch (noun), bit (verb).

4.4 Entailment Recognition

After applying the previous TAC system and the event extraction algorithm, we obtain a new representation for each input **T-H** pair. Instead of computing the surface string similarity, we now compare two pairs of temporal expressions and their corresponding events. Such pairs are defined as *EventTimePairs* (ETPs), and each of them consists of a temporal expression and a noun or a verb denoting the corresponding event. In order to resolve the relation between two ETPs, we need to consider the relation between temporal expressions and extend the results into the whole events. In the following, we first introduce the relations between two temporal expressions, then the lexical resources we have applied, and finally the complete entailment rule representation.

4.4.1 Relations between Temporal Expressions

Relations between temporal expressions have been discussed a lot by researchers. In particular, TimeML has proposed 13 relations to indicate relations between temporal expressions or a temporal expression and an event. For our purpose, three of them are related to the entailment relation. The different granularities and types of temporal expression pairs are also taken into consideration. The relations between a temporal expression and an event are ignored for the moment, since complex lexical semantics may play a role there. Consequently, the possible (entailment) relations between two temporal expressions are shown in Figure 4.1.

	P → P	P → D	D → P	D → D
Same	IDENTITY	NO	INCLUDE	INCLUDE
F → C	INCLUDED	NO	INCLUDE	INCLUDE
C → F	NO	IDENTITY	INCLUDE	INCLUDE

Table 4.1: Relations between temporal expressions

P refers to *time points*, D refers to *duration*, F and C refer to fine and coarse granularity respectively. NO means there is no entailment relation between the two expressions; INCLUDE and INCLUDED indicate the different directions; and IDENTITY is the bi-directional equivalence. For example, both "Oct. 24th, 1997" and "1997" are time points, but the former is finer-grained than the latter. Therefore, the former is INCLUDED

in the latter. While "from 1997 to 1999" is a duration and it is at the same level of granularity as "1997", the former INCLUDES the latter.

4.4.2   Entailment Rules between Events

In order to acquire the relation between two events, we need lexical resources to discover the relations between verbs and nouns, and then combine the results with the relations between temporal expressions.

Lexical Resources   We denote the nouns and verbs corresponding to the temporal expressions as *event types*. The relations between the event types can be determined via lexical resources. WordNet has been widely applied to the RTE task, which is used to discover semantic relations between nouns, e.g., the hypernym/hyponym relation. In our approach, two other features provided by WordNet are considered: 1) the derived form of a noun or a verb; and 2) *entailment* or *entailed-by* relation of a verb. In **H**s, event types are usually represented by verbs, except for those cases where the verb *be* is recognized as the main predicate. To improve the coverage of the verbs in WordNet, we also use VerbOcean (Chklovski and Pantel, 2004) to detect verb relations. In practice, we treat *happens-after*, *stronger-than*, and *similar-to* relations together with the equal relation as monotonic to the entailment relation. The procedure is as follows:

- Verbalize all the nouns, i.e., convert all the nominalizations back to the original verb forms, e.g., "election" to "elect", "winning" to "win".

- Detect possible relations between verbs, e.g., "win" *happens-after* "contest".

- If at least one above-mentioned relation exists, the entailment between event types holds; otherwise, it does not hold.

Rule Representation   In Table 4.2, we define the relations between a pair of ETPs depending on the relations between temporal expressions and between event types.

Even if both event types and temporal expressions have the entailment or inclusion relation, other factors can still change the entailment

	Event Type: YES	Event Type: NO
Temporal Expression: YES	Unknown	NO
Temporal Expression: NO	NO	NO

Table 4.2: Entailment rules between ETPs

between two ETPs, e.g., the different participants of the events. "Unknown" is passed to the later stages. The other three cases determine the false entailment relation[8]. Once entailment relations between ETPs in a sentence are found, these relations can be combined so as to determine the entailment relation between texts, i.e., **T** and **H**. Thus, if the entailment does not hold for all of the ETP pairs, it does not hold for the **T**-**H** pair either; otherwise it is unknown.

To make the process more efficient, we start from **H** to **T**, which is the opposite direction of the entailment relation (Wang and Neumann, 2007a). The motivations are: **H** is the target we need to examine; and **H** is usually simpler than **T**.

Consider the example above again, from **H** we can extract an ETP, "<bit, 1996>". In most cases, the event in **H** is represented by a verb, except for sentence like "The election was in 1992". To deal with such cases, we manually construct a stop word list containing all the forms of the verb *be*. Together with the ETPs extracted from **T** (shown in Section 4.3), we can compare the following pairs of ETPs:

- <release, 1995>, <bit, 1996> $\longrightarrow$ NO

- <win[9], 1996>, <bit, 1996> $\longrightarrow$ NO

- <rematch, 1997>, <bit, 1996> $\longrightarrow$ NO

- <bit, 1997>, <bit, 1996> $\longrightarrow$ NO

Therefore, in this **T**-**H** pair, **T** does not entail **H**.

To sum up, the assumption here is that if all the ETPs of **T** do not entail all the ETPs in **H**, the entailment does not hold between **T** and **H**; otherwise, the answer depends on other information. However, in the

---

[8]In fact, the monotonicity issue is ignored here. The composition of different elements involved in one event highly depends on the event type, which may change the direction of the entailment relation. For instance, "next wednesday" entails "next week", but "I won't come next Wednesday" does not entail "I won't come next week". Nevertheless, for the moment, we simplify it with the intersection of elements.

[9]After applying lexical resources to change the nominalization back into the original verb form.

current system we simplify this problem and consider the latter cases as YES as well.

4.5 Experiments

In this section, we present the evaluation on our system described above. We firstly introduce the datasets we use, and then present the experiments and their results focusing on different aspects. Finally, a detailed error analysis on a subset of the data is given. For the evaluation metrics, we just follow the official RTE challenges[10], i.e., the percentage of matching judgments (system outputs vs. gold-standards) provides the accuracy of the run, i.e., the fraction of correct responses.

4.5.1 Datasets

For the datasets, we extract a subset of the RTE-2[11] and RTE-3[12] datasets. The following two tables summarize information about the datasets.

Corpora	RTE-2		RTE-3		TREC2003	ALL
	dev	test	dev	test		
Both	87 (10.89%)	76 (9.50%)	72 (9.00%)	58 (7.25%)	34 (10.86%)	327 (**8.36%**)
OnlyT	255	291	275	275	100	1196
OnlyH	15	2	10	8	3	38
Neither	442	431	443	459	176	1951
Total	799	800	800	800	313	3912

Table 4.3: Occurrences of the temporal expressions in the datasets

Table 4.3 shows the numbers of **T-H** pairs containing temporal expressions either in both **T** and **H**, only in **T**, only in **H**, or in neither of them. Table 4.4 calculates the frequency of time points and durations.

In addition, we also semi-automatically constructed an additional dataset from TREC2003[13]. The questions and corresponding answers have been used for constructing **H**s and the supporting documents for **T**s. For instance, we combine the question, "What country made the Statue of

---

[10]http://pascallin.ecs.soton.ac.uk/Challenges/RTE3/Evaluation/
[11]http://www.pascal-network.org/Challenges/RTE2
[12]http://www.pascal-network.org/Challenges/RTE3
[13]http://trec.nist.gov/pubs/trec12/t12_proceedings.html

Corpora	RTE-2		RTE-3		TREC2003	ALL
	dev	test	dev	test		
Time point	191	195	209	155	86	836
(per pair)	(2.20)	(2.57)	(2.90)	(2.67)	(2.53)	(2.56)
Duration	37	18	15	12	4	86
(per pair)	(0.43)	(0.24)	(0.21)	(0.21)	(0.12)	(0.26)

Table 4.4: Frequency of different types of temporal expressions in the datasets

Liberty?" and the answer "France" into a statement as **H**, "France made the Statue of Liberty". **T** can take the (ir)relevant documents, e.g., "In 1885, Statue of Liberty arrives in New York City from France". In all, we have constructed 313 **T**-**H** pairs (also shown in Table 4.3 and Table 4.4).

4.5.2   Results

We set up several experiments to evaluate different aspects of our TACTE system. The dependency parser we use is the Stanford Parser (Klein and Manning, 2003). And the following two tables in this subsection show the results.

Corpora	RTE-2		RTE-3		TREC2003	Average
	dev	test	dev	test		
BoW	28.74%	46.05%	40.28%	41.38%	26.47%	37.31%
TACTE	**77.01%**	**68.42%**	**61.11%**	**65.52%**	**64.71%**	**68.20%**
No LexRes	74.71%	67.11%	61.11%	63.79%	52.94%	65.75%

Table 4.5: Experiment results on covered data containing temporal expressions

	RTE-2	RTE-3
BoW (Baseline)	57.88%	61.13%
TACTE + BoW (feature)	58.25%	61.25%
TACTE + BoW (rule)	60.00%	62.88%

Table 4.6: Experiment results on the complete datasets: training on the development set and testing on the test set

In the first experiment, we compare our system with a Bag-of-Words (BoW) system on the data set we extract (Table 4.5). The BoW approach assigns a similarity score to each **T**-**H** pair by calculating the

ratio between the number of overlapping words in **T** and **H** and the total number of words in **H**. Later, a machine learning method SMO (Platt, 1998) in Weka (Witten and Frank, 1999) is used to perform a binary classification[14]. This approach is shown to be a very strong baseline for the RTE task on the current datasets.

Compared with the BoW baseline system performance on the complete datasets (the first row in Table 4.6), the low accuracy shown in the first row in Table 4.5 indicates that the **T-H** pairs containing temporal expressions are more difficult (for the BoW approach). The large improvements (appoximately 21% to 49% on different datasets) of the TACTE system shows the advantage of our strategy combining temporal expression anchoring with event extraction.

In order to find the contribution of the lexical resources, we turn off this part and the third row in Table 4.5 shows the results. It turns out that the lexical resources do not contribute a lot to the whole system. The largest improvement is on the TREC2003 data set, which is the smallest dataset. As an average, this part improves the system with about 2.5% accuracy. The reason is that in these **T-H** pairs with temporal expressions, the respective events in **T** and **H** are easily distinguished. The limited coverage of our lexical resources is another reason. More work on the lexical semantics is necessary, which corresponds to the results of other approaches, e.g., de Marneffe et al. (2006).

We also try to integrate a BoW system into our TACTE system, and there are two ways: either we leave the BoW system to deal with those **T-H** pairs where at least one of the texts does not contain temporal expressions, or the output of our main system is taken as an additional feature in the machine learning procedure. The feature for the latter case is a ternary value: YES, NO, or UNKNOWN. Table 4.6 shows the results of the systems training on the development sets and testing on the test sets.

Since the **T-H** pairs with temporal expressions only cover a small proportion (8.36% in Table 4.3) of the complete data set, the improvement on the complete data set is less obvious. The results in second row is almost the same as the baseline, meaning that a systematic feature selection is necessary for the machine learning approach.

---

[14]In order to keep consistency, we use this classifier for most of our experiments in this dissertation, though here it is a simple threshold learning for the BoW score.

4.5.3 Error Analysis

In this part, we give a detailed error analysis of one of our datasets, i.e., a subset of the RTE-2 development set containing temporal expressions in both **T** and **H**. This subset contains 87 **T-H** pairs, and the TACTE system correctly recognizes 67 pairs. Table 4.7 gives the "error distribution" of the 20 incorrect pairs.

	Errors	Percentage
Extraction	1	5%
Anchoring	2	10%
Parsing	5	25%
Event Extraction	3	15%
Lexical Resources	3	15%
Others	6	30%

Table 4.7: Error distribution

The first kind of errors containing three **T-H** pairs is due to TAC. One error is from SProUT which recognizes "Today" in "USA Today" as a temporal expression. Such an error leads to the false trigger of our anchoring system. Another two errors are implicit temporal expressions introduced by relative clauses and gerunds. In the example "an incident in 1997, when an enraged Mike Tyson bit Holyfield's ear", the relative clause introduced by "when" implies that the "bit" event occurs in the same year as "1997". However, such features cannot be captured and used by our current TAC.

The second kind of errors is due to the RTE system, which contains two subgroups, the parsing part and the event extraction part. We do not discuss the parsing part, since it is out of this dissertation's scope. All of the three errors coming from the event extraction part are due to the wrong selection of the corresponding events. We also tried to extract more possible events, but it resulted in more ambiguity and the performance decreased. For example, in one **T-H** pair, **T** says "...after his landslide victory in Sunday's presidential election", and **H** hypothesizes that person has won the "Sunday's presidential election". Although it is correct to relate "Sunday" with "election", the key events here concerning the entailment relation are "victory" and "won".

Lexical resources also bring errors. For instance, there is no relation found between "was founded" and "was opened". Another example is

the lack of relation between "occur" and "fall on" in the example that "the Chinese New Year occurred on" some day entails "the Chinese New Year's Day falls on" that day.

For the last kind of errors we have not found straightforward solutions yet. Some examples contain complex lexical semantics, e.g., someone "gave up his throne" entails he "abdicated". Another more difficult example is that "the blast observed on Dec. 27 came from ..." entails "the December burst came from ...". Not only the lexical relation between "blast" and "burst" needs to be known, but also "observed" implies that the following event (i.e., "came") happens at the same time as the obsevation.

In short, improvement on the parsing results and lexical resources can solve 40% of the current errors, the remaining part needs more knowledge.

4.6   Related Work

A number of systems with similar goals as TAC have been developed. The semantic tagging system presented by Schilder and Habel (2001) tries to anchor both time-denoting expressions and event-denoting expressions in German news messages. Since event-denoting expressions are more difficult to detect and anchor, the authors admit that only a small set of such expressions can be solved. Han et al. (2006) presented a temporal expression anchorer (TEA), which anchors the temporal expressions in English text and tries to capture their intended meanings. The TEA system was tested on an email dataset with about 150 emails and 279 temporal expressions, and achieves 76.34% accuracy over the test data set.

On the other hand, some researchers working on the RTE task also take temporal expressions into consideration. de Marneffe et al. (2006) extracted and resolved temporal expressions, and used them as features in their approach. However, their system performance barely decreased when these features were excluded. This is consistent with our results mentioned in the second row of Table 4.6. Hickl et al. (2006) also used temporal expressions as features in a machine learning approach. However, there was no separate evaluation showing how much those features contributed to the final results. Tatu et al. (2006) integrated temporal axioms in their rule-based logic inference system. To some extent, these axioms are similar to the different relations between temporal expres-

sions (Section 4.4.1). Whereas the pure rule-based system lacks robustness, when not combined with a statistical backup strategy, our TACTE system first concentrates on those cases containing temporal expressions, and then deals with the whole RTE problem in a more systematic way.

4.7 Extension of the System

As we mentioned in Chapter 3, the main advantage of this extensible architecture is to incorporate multiple specialized modules to handle different cases of entailment, and the main criterion for a good module is its high precision. The TACTE module improves the baseline by a large margin, although the coverage is limited. One natural extension is to consider other types of NEs to improve the coverage. Furthermore, in more complex cases, temporal expressions can only convey one aspect of the whole message, which should be combined with other information obtained. In the following, we firstly present the extended system architecture and then the evaluation results together with some discussion.

4.7.1 Extended System Architecture

Figure 4.4 shows the architecture of the extended system. We mainly extend the temporal expressions into other NE types, i.e., person names, location names, and organization names. The process is quite similar to the TACTE system. Accordingly, the ETP can be extended into the following *Event Tuple* (ET),

- *<EventType, Time, Location*, List*<Participants>>*

Event Type can be either a noun or a verb; *Time* is a normalized temporal expression; *Location* is a location name; a *Participant* can be either a person name or an organization name. In particular, after referring several geographic taxonomies, Geonames[15], WorldGazetteer[16], and so on, we construct a geographic ontology using geographic terms and two relations. The backbone taxonomy of the ontology is shown in the following Figure 4.5.

The taxonomy consists of geographic terms referring different granularities of areas. Inside each *Country*, we have two categories of fine-grained

---

[15]Geonames geo coding web service: `http://www.geonames.org/`
[16]WorldGazetteer: `http://www.world-gazetteer.com`
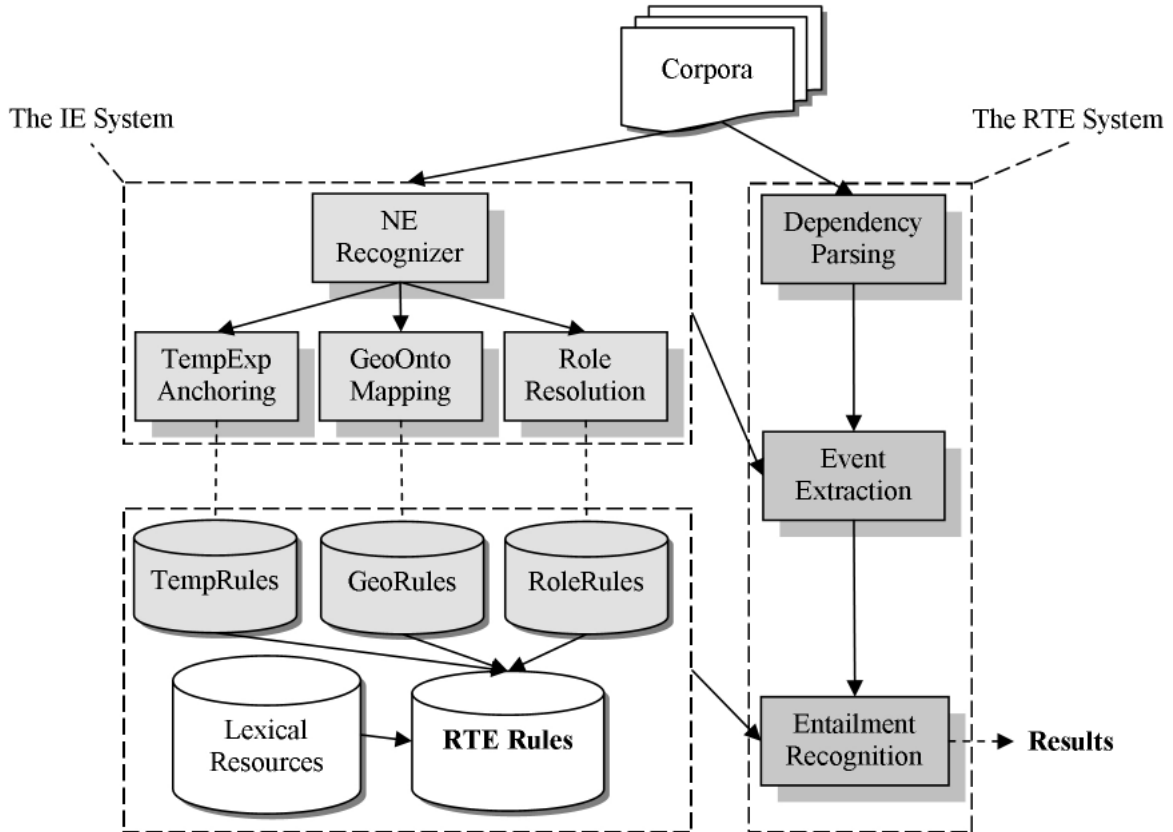
Figure 4.4: Architecture of the extended TACTE system.

places, i.e., artificial divisions and natural places. The basic relation between the terms is the directional *part-of* relation, which means the geographic area on the right side is contained in the area on the left side. In addition, extra geographic areas are connected with these basic terms using the same part-of relation. For example, the following geographic areas consist of the basic terms above:

- **Subcontinent**: *the Indian subcontinent, the Persian Gulf*, etc.

- **Subcountry**: *Lower Saxony, the Western USA*, etc.

An additional *equal* relation is utilized for synonyms and abbreviations of the same geographic area, e.g., "the United Kingdom", "the UK", "Great Britain", and so on. Consequently, the entailment rules between ETs also have more dimensions. In summary, all the information contained in **H** must be fully entailed by **T**; otherwise, it is No[17].

---

[17]Being similar to the TACTE system, we also intersect all the relations between elements of two ETs, and leave the monotonicity problem for the future.

Planet
(i.e. Earth) ←*part-of*—Continent←*part-of*—Country ←*part-of*—City/Town/... (artificial)
←*part-of*—River/Island/... (natural)
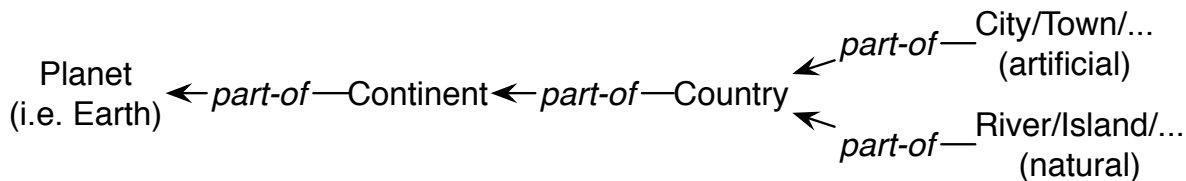
Figure 4.5: The backbone taxonomy of the geographical ontology

4.7.2 Experiments

In this section, we present our evaluation of the system in the context of the TAC 2008 RTE Track[18]. In order to cover the whole RTE dataset, we incorporate three specialized modules into our system, the TACTE module, the NE-M module (Section 4.7.1), and another specialized module developed before, called TS-M (explained below). In addition, two simple baseline methods are used as fallbacks, BoW-B (based on bag-of-words similarity) and Tri-B (based on bag-of-dependency-triples similarity). More details about the backup modules can be found in the last part of Section 3.2.

The TS-M module implements the main approach proposed by Wang and Neumann (2007a). The main idea is to extract a new sentence representation called *Tree Skeleton* (TS) based on the dependency parse trees, and then use a kernel-based machine learning method to make the prediction of the entailment relation. The TS structure can be viewed as an extended version of the predicate-argument structure. Since it contains not only the predicate and its arguments, but also the dependency paths between them, it captures the essential part of the sentence. We utilize the subsequence kernel (Bunescu and Mooney, 2006) to represent the differences between two TSs. More details can be found in the next chapter (Section 5.4.2) as well as the original paper (Wang and Neumann, 2007a).

We have submitted three configurations of our system for the challenge, which differ in assignment of different weights to the used RTE-modules. According to the performances of the modules on the development sets, the voting model is simply taking the results from the module which has the highest accuracy. Those pairs that are not covered by any specialized modules are passed to the fallback modules.

The configurations of the three submissions for the two-way task and the results are as follows:

---

[18]http://www.nist.gov/tac/tracks/2008/rte/

- Run1: TACTE, TS-M, and Tri-B;

- Run2: TACTE, TS-M, and BoW-B;

- Run3: TACTE, TS-M, NE-M, and Tri-B, BoW-B.

Settings	Yes (500)	No (500)	All (1000)
Run1	66.6%	67.8%	67.2%
Run2	81.4%	58.4%	69.9%
Run3	74.8%	66.4%	**70.6%**

Table 4.8: Performance of the whole system (two-way)

Since our modules are not specially designed for recognizing three-way entailment, we take a strategy to combine results from different modules. For specialized modules, we keep YES as ENTAILMENT, but change NO into UNKNOWN. For the backup modules, we take the following rules:

- If BoW-B=YES & Tri-B=NO then CONTRADICTION;

- If BoW-B=YES & Tri-B=YES then ENTAILMENT;

- Others UNKNOWN.

The configurations of the three submissions for the three-way task and the results are:

- Run1: TAC-M, TS-M, and Tri-B, BoW-B;

- Run2: TAC-M, TS-M, NE-M (partial), and Tri-B, BoW-B;

- Run3: TAC-M, TS-M, NE-M, and Tri-B, BoW-B.

Settings	Entailment (500)	Contradiction (150)	Unknown (350)	All (1000)
Run1	68.2%	38.7%	61.4%	**61.4%**
Run2	66.6%	41.3%	47.1%	56.0%
Run3	72.8%	33.3%	54.9%	60.6%

Table 4.9: Performance of the whole system (three-way)

Comparing the two-way task and the three-way task, we find that CONTRADICTION cases are not trivial to capture (with only around 40%

accuracy), the difficulty and importance of which are also discussed by de Marneffe et al. (2008).

Compared with other RTE systems, our system ranked the 3rd place for both two-way and three-way evaluation in the TAC 2008 RTE track (Giampiccolo et al., 2009). Our extensible architecture together with specialized modules looks very promising. In order to see more details of each specialized modules, we break down the results in the following discussion.

4.7.3 Discussion

Table 4.10 shows the performance of each specialized module in the two-way evaluation.

Tasks	TACTE	NE-M	TS-M	BoW-B	Tri-B
IR(300)	75.0% (4)	61.0% (164)	76.5% (85)	63.3%	54.3%
QA(200)	90.0% (10)	54.8% (93)	73.2% (82)	49.0%	53.5%
SUM(200)	83.3% (6)	55.2% (67)	74.5% (51)	63.5%	54.0%
IE(300)	72.7% (11)	46.7% (152)	74.2% (128)	50.0%	50.0%
All(1000)	**80.6% (31)**	**54.3% (477)**	74.6% (346)	56.5%	52.8%

Table 4.10: Accuracy and coverage of each RTE module

The TACTE has the highest accuracy, though the coverage is the lowest. The performance of TS-M is also higher than the average accuracy. The NE-M module does not have a good accuracy, which may be caused by the lower performance of recognizing other types of NEs (compared with temporal expressions).

Bobrow et al. (2007) also proposed a precision-oriented approach, however with a much lower coverage on the whole data set. MacCartney and Manning (2007) applied natural logic to the RTE task, and also dealt with specific cases of entailment pairs, e.g., quantifiers. Many other approaches explored the limitation of coverage, e.g., using lexical-syntactic rules (Bar-Haim et al., 2007). This confirms that the RTE task cannot be easily solved by using only one single generic method, but the combination of different approaches.

Bos and Markert (2005) combined a rigid logic inference system with shallow lexical features to gain from both sides. MacCartney and Manning (2007) applied a shallow system in order to achieve the full coverage of the data set, which is similar to our backup modules. Our strategy on

this aspect is to rank the different modules based on their performance on the development data, so that a high precision is maximally preserved.

4.8   Summary

In this chapter, we firstly present one specialized RTE module which deals with text pairs containing temporal expressions. After extracting and anchoring the temporal expressions, our TACTE system takes them as starting points in dependency structures and searches for events corresponding to these expressions. With the help of the entailment rules (including lexical resources), the entailment relation can be detected. Several experiments on various datasets are conducted, and TACTE shows a significant improvement over the baseline system.

Then, we show the extension of the TACTE system into other types of NEs and utilize a unified event representation. Apart from the separate evaluation on the module, we also perform experiments with the whole RTE system consisting of several specialized modules. The results on the RTE challenge data show the advantage of such extensible architectures. Although the coverage of each specialized module is limited, the high precision is the key requirement. Our result is quite consistent with other researchers' work, and it seems to indicate an effective way of handling this challenging task.

In order to combine all modules' results, we rank the modules with weights that have been automatically derived from a performance analysis using training data. The voting strategy can be further explored in the future to achieve a better picture of which entailment cases can be more reliably handled by which RTE module.

The error analysis also shows some complex cases of entailment, which require sophisticated reasoning and/or (external) commonsense knowledge. In the next chapter, we focus on this part and introduce our work on both applying inference rules to the RTE system and refining such rule collections.

5 Textual Entailment with Inference Rules

In this chapter[1], we present our work on applying inference rules to the RTE task. We extend and refine an existing inference rule collection using a hand-crafted lexical resource. In order to accurately discover the text pairs to trigger the rules, we preprocess each sentence and extract a dependency-path-based representation. The experimental results demonstrate that this is another precision-oriented approach, which can also be viewed as a specialized module. The coverage of this module highly depends on the external knowledge base, i.e., the inference rule collection. Addressing the problem of resource creation, we also present a pilot study on acquiring paraphrased fragment pairs in an unsupervised manner. Such resources can be potentially useful for entailment recognition as well as for other tasks.

---

[1]Section 5.1 to Section 5.5 have been published in (Dinu and Wang, 2009), and it was a collaboration with Georgiana Dinu, who contributed mostly to the refinement of the inference rule collection.

5.1   Overview

Studies such as Clark et al. (2007) attest that lexical substitution (e.g., of synonyms or antonyms) or simple syntactic variation account for entailment recognition only in a small number of pairs. Thus, one essential issue is to identify more complex expressions which, in appropriate contexts, convey the same (or similar) meaning. However, more generally, we are also interested in pairs of expressions in which only a uni-directional inference relation holds[2].

A typical example is the following RTE pair in which "accelerate to" in **H** is used as an alternative formulation for "reach speed of" in **T**.

T: *The high-speed train, scheduled for a trial run on Tuesday, is able to* **reach** *a maximum* **speed of** *up to 430 kilometers per hour, or 119 meters per second.*

H: *The train* **accelerates to** *430 kilometers per hour.*

One way to deal with textual inference is through rule representation, for example *X wrote Y ≈ X is author of Y*. However, manually building collections of inference rules is time-consuming and it is unlikely that humans can exhaustively enumerate all the rules encoding the knowledge needed in reasoning with natural language. Instead, an alternative is to acquire these rules automatically from large corpora. Given such rule collections, the next step to focus on is how to successfully use it in NLP applications. We consider both aspects, inference rules and using them for the RTE task.

5.2   Inference Rules

A number of automatically acquired inference rule/paraphrase collections are available, such as Szpektor et al. (2004) and Sekine (2005). In our work, we use the DIRT collection (Lin and Pantel, 2001), because it is the largest one available and it has a relatively good accuracy (in the 50% range for top generated paraphrases (Szpektor et al., 2007)).

The DIRT collection of inference rules has been acquired based on the *Extended Distributional Hypothesis*. The original *Distributional Hypothesis* (DH) (Harris, 1954) states that *words* occurring in similar contexts

---

[2]We use the term inference rule to stand for this concept; the two expressions can be actual paraphrases if the relation is bi-directional.

have similar meaning, whereas the extended version hypothesizes that *phrases* occurring in similar contexts are similar. One of the main advantages of using the DH is that the only input needed is a large corpus of (parsed) text. Another line of work on acquiring paraphrases uses comparable corpora, for instance, Barzilay and McKeown (2001) and Pang et al. (2003), and we come back to this point in our pilot study presented in the end of this chapter (Section 5.6).

An inference rule in DIRT is a pair of binary relations[3] $\langle pattern_1(X, Y),$ $pattern_2(X, Y) \rangle$ which stands for an inference relation. $Pattern_1$ and $pattern_2$ are paths in dependency trees obtained with the Minipar parser (Lin, 1998), while X and Y are placeholders for nouns at the end of this chain. The two patterns constitute a candidate paraphrase if the sets of X and Y values exhibit relevant overlap. In the following example, the two patterns are *prevent* and *provide protection against.*

$$\mathbf{X} \xleftarrow{subj} prevent \xrightarrow{obj} \mathbf{Y}$$

$$\mathbf{X} \xleftarrow{subj} provide \xrightarrow{obj} protection \xrightarrow{mod} against \xrightarrow{pcomp} \mathbf{Y}$$

Such rules can be informally defined as directional relations between two text patterns with variables (Szpektor et al., 2007). The left-hand-side pattern is assumed to entail the right-hand-side pattern in certain contexts, under the same variable instantiation. The definition relaxes the intuition of inference, as we only require the entailment to hold in *some* but not *all* contexts, motivated by the fact that such inferences occur often in natural text.

Based on observations of using the inference rule collection on real data, we discover that:

1. Some of the needed rules are still missing even in a very large collection such as DIRT.

2. There are some systematically erroneous rules in the collection can be excluded.

We address both of these problems with a hand-crafted lexical resource, i.e., WordNet (Fellbaum, 1998). Table 5.1 gives a selection of such rules. Notice that the dependency relations are omitted in the representation for convenience.

---

[3]For simplification, we take binary relations as examples, but in principle, each relation can contain more (or less) components.

X write Y →X author Y
X, founded in Y →X, opened in Y
X launch Y → X produce Y
X represent Z   → X work for Y
death relieved X → X died
X faces menace from Y ↔ X endangered by Y
X, peace agreement for Y   *→ X is formulated to end war in Y*

Table 5.1: Example of inference rules needed in RTE

The first rows contain rules which are structurally very simple. These, however, are missing from DIRT and most of them also from other hand-crafted resources such as WordNet, i.e., there is no short path connecting them in the semantic network of words. This is to be expected as they are rules which hold in specific contexts, but are difficult to be captured by a sense distinction of the lexical items involved. For example "launch" entails "produce" when the context is that of a company launching a new line of products.

The more complex rules are even more difficult to capture with a DIRT-like algorithm. Some of these do not occur frequently enough even in large amounts of text to permit acquiring them via the DH[4].

5.3   Combining DIRT with WordNet

We use WordNet to augment the original inference rule collection of DIRT and exclude some of the incorrect rules. In order to address the issue of missing rules, we extend the DIRT rules by adding rules in which any of the lexical items involved in the patterns can be replaced by WordNet synonyms.   In the example above, we consider the DIRT rule *X face threat of Y → X, at risk of Y* (Table 5.2).

Naturally, due to the lack of sense disambiguation, our method introduces many rules that are incorrect. As one can see, expressions such as "front scourge" do not make any sense, therefore any rules containing this expression are incorrect. However, some of the new rules created in this example, such as *X face threat of Y ≈ X, at danger of Y* are

---

[4]For example a Google search for "face menace from" yields less than ten hits (November, 2010).

X face threat of Y
    ≈ X, at risk of Y
___

face
    ≈ confront, front, look, face up

threat
    ≈ menace, terror, scourge
___

risk
    ≈ danger, hazard, jeopardy,
    endangerment, peril

Table 5.2: Lexical variations creating new rules based on DIRT rule *X face threat of Y → X at risk of Y*

reasonable and the incorrect rules often contain patterns that are very unlikely to occur in natural text.

The idea behind this is that a combination of different lexical resources is needed in order to cover the vast variety of phrases which humans can judge to be in an inference relation. The method just described allows us to identify the first three rules listed in Table 5.1. For example *X opened in Y ≈ X founded in Y* is added because *X opened in Y ≈ X launched in Y* is a DIRT rule and *launch* and *found* are synonyms in WordNet. We also acquire the rule *X face menace of Y ≈ X endangered by Y* (via *X face threat of Y ≈ X threatened by Y*, *menace ≈ threat*, *threaten ≈ endanger*).

In our experiments, we also make a step towards removing the most common systematic errors present in DIRT. A fundamental weakness of the DH algorithms is that not only phrases with the same meaning are extracted but also phrases with *opposite* meanings. In order to overcome this problem (and since such errors are relatively easy to detect), we applied a filter to the DIRT rules, which eliminates inference rules that contain WordNet antonyms. For such a rule to be eliminated, the two patterns have to be identical (with respect to edge labels and content words) except from the antonymous words; an example of a rule eliminated this way is *X **have** confidence in Y ≈ X **lack** confidence in Y.*

As pointed out by Szpektor et al. (2007), a thorough evaluation of a rule collection is not a trivial task; however, due to our methodology, we can assume that nearly all rules eliminated this way are indeed erroneous. We evaluate our extension and filtering of the DIRT rules by their effect

on the RTE task instead (Section 5.5).

5.4   Applying Inference Rules to RTE

In this section, we focus on applying the inference rule collection to the RTE task. Firstly, we point out two issues that are encountered when directly applying inference rules to the **T-H** pairs. One issue is concerned with correctly identifying the text pairs, in which the knowledge encoded in these rules is needed. Following that, another non-trivial task is to determine the way this knowledge interacts with the rest of the information conveyed in the text pair. In order to further investigate these issues, we apply the rule collection on a dependency-based representation of **T** and **H**, namely a *Tree Skeleton* (TS) derived from the parse trees, which aims to capture the essential information conveyed by **T** and **H**.

5.4.1   Observations

A straightforward experiment can reveal the number of pairs in the RTE data which can be covered by the DIRT rules. For all the experiments in this chapter, we use the DIRT collection provided by Lin and Pantel (2001), derived from the DIRT algorithm applied on 1GB of news text. The results we report here use only the most confident rules amounting to more than 4 million rules (TOP 40 following Lin and Pantel (2001))[5].

Assuming that $\langle pattern_1(X,Y), pattern_2(X,Y) \rangle$ is an inference rule, we identify RTE pairs in which $pattern_1(w_1, w_2)$ and $pattern_2(w_1, w_2)$ are matched, one in **T** and the other in **H**. Such a matching is performed after dependency parsing, and all the words are lemmatized. This is called a successful *rule application* all through this chapter. However, on average, only 2% of the pairs in the RTE data is subject to the application of such inference rules. Out of these, approximately 50% are lexical rules (one verb entailing the other); and out of these lexical rules, around 50% are present in WordNet in a synonym, hypernym or sister relation. At a manual analysis, close to 80% of these are correct rules; this is higher than the estimated accuracy of DIRT, probably due to the bias of the data which consists of pairs which are candidates of positive entailment relation.

---

[5]Another set of experiments show that for this particular task, using the entire collection instead of a subset gives similar results.

Given the small number of inference rules identified in this way, we performed another analysis. The second analysis aims at determining the upper bound of the number of pairs in the RTE corpora, which the inference rules can be applied to. We compute in how many pairs the two patterns of an inference rule can be matched irrespective of their variable values. Altogether in around 20% of the pairs, patterns of a rule can be found, many times with more than one rule matching a pair. However, notice that in many of these pairs, finding the patterns of an inference rule does not imply that the rule is truly applicable to that pair.

To sum up, making use of the knowledge encoded in inference rules is not a trivial task. If rules are used strictly in concordance with their definition, their utility is limited to a very small number of entailment pairs. For this reason:

1. Instead of forcing the variable values to be identical as most previous work, we allow more flexible rule matching (similar to Marsi et al. (2007)).

2. Furthermore, we control the rule application process using a text representation based on dependency structure, i.e., the tree skeleton (Section 5.4.2).

Even if a system is capable of correctly identifying the cases in which an inference rule is applicable, subsequent issues arise from the way these fragments of text interact with the surrounding context. Assuming we have a correct rule present in an entailment pair, the cases in which the pair is still not a positive case of entailment can be summarized as follows:

- The inference rule is matched in the text, but it is either a partial match or embedded in other predicates/modifiers which block the entailment, e.g., negative markers, modifiers, embedding verbs not preserving entailment[6].

- The rule is correct in a limited number of contexts, but the current context is not the correct one.

In order to investigate these issues, we choose to apply the rule collection on a dependency-based representation of **T** and **H**. We firstly introduce this representation and the algorithm to derive it, and following that we describe how we apply the inference rules on this structure.

---

[6]See (Nairn et al., 2006) for a more detailed analysis of these aspects.

5.4.2   Tree Skeleton

The *Tree Skeleton* (TS) representation was proposed by Wang and Neumann (2007a), and can be viewed as an extended version of the predicate-argument structure. Since it contains not only the predicate and its arguments, but also the dependency paths between them, it captures the essential part of the sentence.

Following the algorithm, we first preprocess the data using a dependency parser[7] and then select overlapping topic words (i.e., nouns) in **T** and **H**. By doing so, we use fuzzy match at the substring level instead of full match (Wang and Neumann, 2007a). Starting with these nouns, we traverse the dependency tree to identify the lowest common ancestor node (which we call the *root node*). This sub-tree without the inner yield is defined to be the tree skeleton. Figure 5.1 shows the TS of **T** of the following positive example,

T: *For their discovery of ulcer-causing bacteria, Australian doctors Robin Warren and Barry Marshall have received the 2005 Nobel Prize in Physiology or Medicine.*

H: *Robin Warren was awarded a Nobel Prize.*
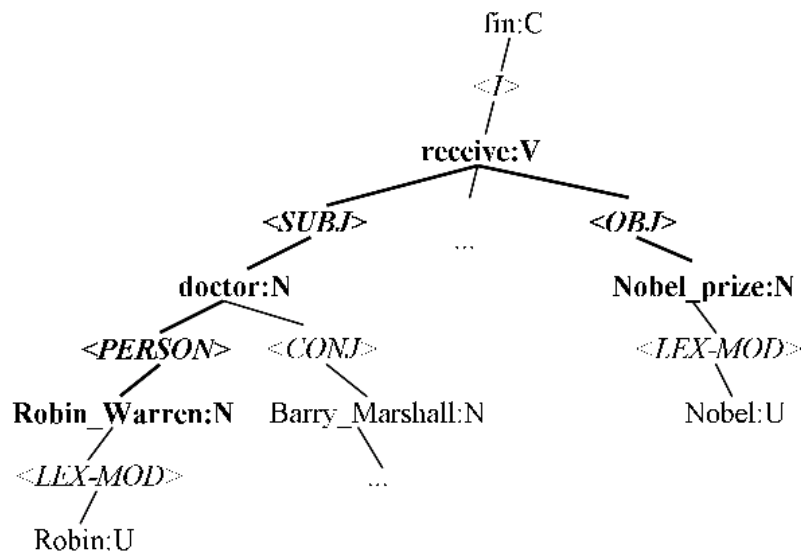
Figure 5.1: The dependency structure of the text (tree skeleton in bold)

Notice that in order to match inference rules with two variables, the number of dependency paths contained in a TS should also be two. In

---

[7]Here we use Minipar for consistency with the DIRT collection.

practice, among all the 800 **T-H** pairs of the RTE-2 test set, we successfully extracted tree skeletons in 296 text pairs, i.e., 37% of the test data is covered by this step and results on other data sets are similar.

5.4.3 Rule Application

We perform a straightforward matching algorithm to apply the inference rules on top of the tree skeleton structure. Given tree skeletons of **T** and **H**, we check if the two respective left dependency paths, the two right ones or the two root nodes match the patterns of a rule. In the example above, the rule $X \xleftarrow{obj} receive \xrightarrow{subj} Y \approx X \xleftarrow{obj2} award \xrightarrow{obj1} Y$ satisfies this criterion, as it is matched at the root nodes. Notice that the rule is correct only in restricted contexts, in which the object of *receive* is something which is conferred on the basis of merit. In this pair, the context is indeed the correct one.

5.5 Experiments

Our experiments consist of predicting positive entailment in a very straightforward rule-based manner. For each collection, we select the RTE pairs in which we find a tree skeleton and that matches an inference rule. The first number in our table entries represents how many of such pairs we have identified, out the 1600 of development and test pairs. For these pairs we simply predict positive entailment and the second entry represents what percentage of these pairs are indeed positive entailments. This work does not focus on building a complete RTE system; however, we also combine our method with a bag of words baseline to see the effects on the whole data set.

5.5.1 Results on the Covered Dataset

Table 5.3 summarizes the results using three different rule collections.
In the first two columns ($Dirt_{TS}$ and $Dirt+WN_{TS}$), we consider DIRT in its original state and DIRT with rules generated with WordNet (as described in Section 5.3); all precisions are higher than 67%[8]. After

---

[8]The average accuracy of the systems in the RTE-3 challenge is around 61% (Giampiccolo et al., 2007)

Dataset	Dirt_{TS}	Dirt+WN_{TS}	Id_{TS}	Dirt+Id+WN_{TS}	Dirt+Id+WN
RTE-2	69.38% (49)	67.02% (94)	66.66% (45)	65.38% (130)	50.07% (673)
RTE-3	69.04% (42)	70.00% (70)	79.31% (29)	72.05% (93)	55.06% (661)

Table 5.3: Precision on the covered dataset with various rule collections

adding WordNet, approximately in twice as many pairs, tree skeletons and rules are matched, while the precision is not much harmed. This may indicate that our method of adding rules does not decrease precision of an RTE system[9].

In the third column, we report the results of using a set of rules containing only the trivial identity ones (Id_{TS}). For our current system, this can be seen as a precision upper bound for all the other collections, in concordance with the fact that identical rules are nothing but inference rules of highest possible confidence. The fourth column(Dirt+Id+WN_{TS}) contains what can be considered our best setting. In this setting, considerably more pairs are covered using a collection containing DIRT and identity rules with WordNet extension.

Although the precision results with this setting are encouraging (65% for RTE-2 data and 72% for RTE-3 data), the coverage is still low, 8% for RTE-2 and 6% for RTE-3. This aspect together with an error analysis we performed are the focus of Section 5.5.3.

The last column (Dirt+Id+WN) gives the precision we obtain if we simply decide a pair is true entailment when we have an inference rule matched in it (irrespective of the values of the anchors or of the existence of tree skeletons). As expected, only identifying the patterns of a rule in a pair irrespective of tree skeletons does not give any indication of the entailment value of the pair.

5.5.2   Results on the Entire Dataset

Finally, we also integrate our method with a bag of words baseline, which calculates the ratio of overlapping words in **T** and **H**. For the pairs that our method covers, we overrule the baseline's decision. The results are

---

[9]Indeed, sense ambiguity gives rise to lots of incorrect rules; however there seems to be no indication that these incorrect rules appear in the tree skeletons of the two texts, to a greater extent than DIRT incorrect rules.

Dataset	BoW	Main
RTE2 (85 pairs)	51.76%	60.00%
RTE3 (64 pairs)	54.68%	62.50%

Table 5.4: Precision on covered RTE data

Dataset (800 pairs)	BoW	Main & BoW
RTE2	56.87%	57.75%
RTE3	61.12%	61.75%

Table 5.5: Precision on full RTE data

shown in Table 5.5 (*Main* stands for the Dirt + Id + WN$_{TS}$ configuration). On the full data set, the improvement is still small due to the low coverage of our method, however on the pairs covered by our method (Table 5.4), there is a significant improvement over the baseline.

5.5.3 Discussion

In this section, we take a closer look at the data in order to better understand how our method of combining tree skeletons and inference rules works. We firstly perform an error analysis on what we have considered our best setting. Following that, we analyze data to identify the main reasons causing the low coverage.

For error analysis we consider the pairs of the RTE-3 test data set which have been incorrectly classified, consisting of a total of 25 pairs. We classify the errors into three main categories: rule application errors, inference rule errors, and other errors (Table 5.6).

In the first category, the tree skeleton fails to match the corresponding anchors of the inference rules. For instance, if someone founded "the Institute of Mathematics (Instituto di Matematica) at the University of Milan", it does not follow that they founded "The University of Milan".

Source of error	% pairs
Incorrect rule application	32%
Incorrect inference rules	16%
Other errors	52%

Table 5.6: Error analysis of the incorrectly classified text pairs in the RTE-3 test set

A rather small portion of the errors (16%) are caused by incorrect inference rules. Out of these, two are correct in some contexts but not in those **T-H** pairs in which they are found. For example, the following rule *X generate Y* ≈ *X earn Y* is used incorrectly, however in the restricted context of money or income, the two verbs have similar meaning. An example of an "real" incorrect rule is *X issue Y* ≈ *X hit Y*, since it is difficult to find a context in which this holds.

The last category contains all the other errors, most of which require finer-grained analysis of the lexical information, e.g., specific types of adjectives, different classes of modal verbs, and so on.

For the second part of our analysis we discuss the coverage issue, based on an analysis of uncovered pairs. A main factor in failing to detect pairs in which inference rules should be applied is that the tree skeleton does not find the corresponding lexical items of two rule patterns.

Issues occur even if the tree skeleton structure is modified to align all the corresponding fragments together. Consider cases such as "threaten to boycott" and "boycott" or similar constructions with other embedding verbs such as "manage", "forget", "attempt". Our method can detect if the two embedded verbs convey a similar meaning, however, not how the embedding verbs affect the implication.

Independent of the shortcomings of our tree skeleton structure, a second factor in failing to detect true entailment still lies in lack of rules. For instance, the last two examples in Table 5.1 are entailment pair fragments which can be formulated as inference rules, but it is not straightforward to acquire them via the DH.

In the rest of this chapter, we present a pilot study of acquiring paraphrased fragment pairs using monolingual comparable corpora, which can be viewed as an alternative to the DIRT-style rules acquired based on the DH.

5.6   Pilot Study: Paraphrase Acqusition

Paraphrase is an important linguistic phenomenon which occurs widely in human languages. Since paraphrases capture the variations of linguistic expressions while preserving the meaning, they are very useful in many applications, such as machine translation (Marton et al., 2009), document summarization (Barzilay et al., 1999), and recognizing textual entailment (RTE) (Dagan et al., 2005). However, such resources are not trivial to

obtain.

A variety of paraphrase extraction approaches have been proposed recently, and they require different types of training data. Some require bilingual parallel corpora (Callison-Burch, 2008, Zhao et al., 2008), others require monolingual parallel corpora (Barzilay and McKeown, 2001, Ibrahim et al., 2003) or monolingual comparable corpora (Dolan et al., 2004). In this study, we focus on extracting paraphrase fragments from monolingual corpora, because this is the most abundant source of data. Additionally, this potentially allows us to extract paraphrases for a variety of languages that have monolingual corpora, but which do not have easily accessible parallel corpora.

In particular, we address the following issues:

1. Adapting a translation fragment pair extraction method to paraphrase extraction;

2. Construction of a large collection of paraphrase fragments;

3. Manual evaluation of both intermediate and final results of the paraphrase collection.

The focus of this work is on fragment extraction, but we briefly describe document and sentence pair extraction first. We evaluate quality at each stage using Amazon's Mechanical Turk (MTurk)[10].

5.6.1   Document Pair Extraction

Monolingual comparable corpora contain texts about the same events or subjects, written in one language by different authors (Barzilay and Elhadad, 2003). We extract pairs of newswire articles written by different news agencies from the GIGAWORD corpus, which contains articles from six different agencies.

We used *Lucene*'s *MoreLikeThis* function[11], which calculates the number of overlapping words weighting them based on TF-IDF. We found document pairs with >0.9 were classified by annotators to be related more than half the time. We performed subsequent steps on the 3896 document pairs that belonged to this category.

---

[10]`https://www.mturk.com/mturk/`
[11]`http://lucene.apache.org/java/2_9_1/api/contrib-queries/org/apache/lucene/search/similar/MoreLikeThis.html`

5.6.2   Sentence Pair Extraction

After extracting pairs of related documents, we next selected pairs of related sentences from within paired documents. To do so, we selected sentences with overlapping n-grams up to length n=4. Obviously for paraphrasing, we want some of the n-grams to differ, so we varied the amount of overlap and evaluated sentence pairs with a variety of threshold bands. Our best scoring threshold band was 0.2-0.8. Sentence pairs with this overlap were judged to be paraphrases 45% of the time, to be related 30% of the time, and to be unrelated 25% of the time. Although the F2 heuristic proposed by Dolan et al. (2004), which takes the first two sentences of each document pair to be equivalent obtains higher relatedness score (our evaluation showed that among the F2 sentences were 50% paraphrases, 37% related, and 13% unrelated), our n-gram overlap method extracted much more sentence pairs per document pair. We used 276,120 sentence pairs to feed our fragment extraction method.

5.6.3   Fragment Pair Extraction

The basic procedure of fragment pair extraction is to 1) establish alignments between words or n-grams and 2) extract target paraphrase fragments. For the first step, we use two approaches. One is to change the common substring alignment problem from letters to word sequences and we extend the longest common substring (LCS) extraction algorithm (Bergroth et al., 2000) to multiple common n-grams. An alternative way is to use a normal word aligner (widely used as the first step in MT systems) to accomplish the job. For our experiments, we use the BerkeleyAligner[12] (Liang et al., 2006) by feeding it a dictionary of pairs of identical words along with the paired sentences. We can also combine these two methods by performing the LCS alignment first and adding additional word alignments from the aligner. These form the three configurations of our system (Table 5.7).

Following Munteanu and Marcu (2006), we use both positive and negative lexical occurrence probabilities. The positive probability measures how likely one word is aligned to another (value from 0 to 1); and the negative probability indicates how likely there is NO alignment exists between a word pair (from -1 to 0). The basic idea to have both is

---

[12]http://code.google.com/p/berkeleyaligner/
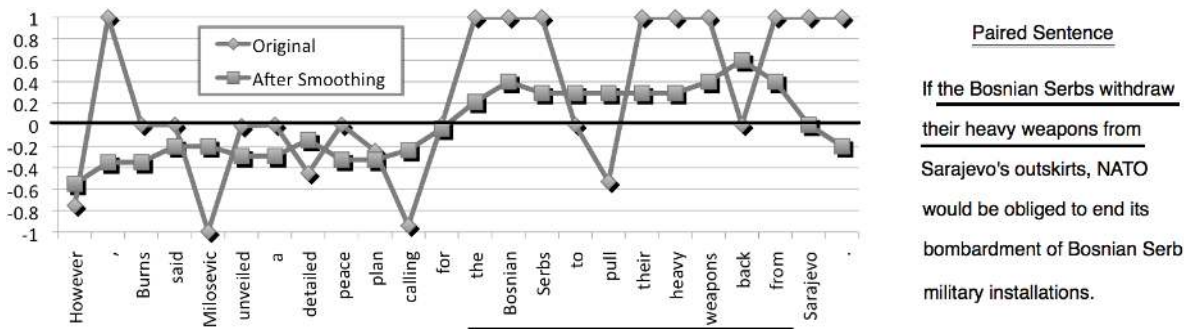
Figure 5.2: An example of fragment pair extraction

that when a word cannot be aligned with any other word, it chooses the *least unlikely* one. If the positive probability of w_1 being aligned with w_2 is defined as the conditional probability $p(w_1|w_2)$, the negative probability is simply $p(w_1|\neg w_2)$[13]. Since we obtain a distribution of all the possible words aligned with w_1 from the word aligner, both $p(w_1|w_2)$ and $p(w_1|\neg w_2)$ can be calculated; for the LCS alignment, we simply set $p(w_1|w_2)$ as 1 and $p(w_1|\neg w_2)$ as -1, if w_1 and w_2 are aligned; and vice versa, if not.

After the initialization of all the word alignments using the two probabilities, each word takes the average of the neighboring four words and itself as its smoothed probability. The intuition of this smoothing is to tolerate a few unaligned parts (if they are surrounded by aligned parts). Finally, all the word alignments having a positive score are selected as candidate fragment elements. Figure 5.2 shows an example of this process[14].

The second step, fragment extraction, is a bit tricky, since a *fragment* is not clearly defined like a *document* or a *sentence*. One option is to follow the MT definition of a *phrase*, i.e., a sub-sentential n-gram string (usually n is less than 10). Munteanu and Marcu (2006) adopted this, and considered all the possible sub-sentential translation fragments as their targets, i.e., the adjacent n-grams. For instance, in Figure 5.2, all the adjacent words above the threshold (i.e., zero) form the target paraphrase, "the Bosnian Serbs to pull their heavy weapons back from" and those aligned words in the other sentence "the Bosnian Serbs withdraw their heavy weapons from" are the source paraphrase. The disadvantage of this definition is that the extracted fragment pairs may not be easy

---

[13]$\neg w_2$ indicates any other word except w_2.

[14]Stop words are all set to 1 initially. Zero is the threshold, and the underscored phrases are the outputs.

for humans to interpret or even be ungrammatical (cf. the 3rd example in Table 5.8). An alternative way is to follow the linguistic definition of a *phrase*, e.g., noun phrase (NP), verb phrase (VP), etc. In this case, we need to use (at least) a chunker to preprocess the text and obtain the proper boundary of each fragment. We used the OpenNLP chunker[15] for this purpose.

We finalize our paraphrase collection by filter out identical fragment pairs, subsumed fragment pairs (one fragment is fully contained in the other), and fragments containing only one word. Apart from sentence pairs collected from the comparable corpora, we also did experiments on the existing MSR paraphrase corpus[16] (Dolan and Brockett, 2005), which is a collection of manually annotated sentential paraphrases.

The evaluation on both collections is done by the MTurk. Each task contains 8 pairs of fragments to be evaluated, plus one positive control using identical fragment pairs, and one negative control using a pair of random fragments. All the fragments are shown with the corresponding sentences from where they are extracted. The question being asked is:

- *How are the two highlighted phrases related?*

The possible answers are:

- *These phrases refer to the same thing as each other.* (PARAPHRASE)

- *These phrases are overlap but contain different information'.* (RELATED)

- *The phrases are unrelated or invalid.* (INVALID)

Table 5.8 shows some examples.

We manually evaluated 1051 sentence pairs in all, and Table 5.7 shows the results (excluding invalid sentence pairs). We use LCS or the word aligner for initialization and apply n-gram-based or chunk-based phrase extraction. The first column serves as the baseline.

In general, the results on MSR is better than those on our corpus[17]. Comparing the different settings, for our corpus, word alignment with n-gram fragment extraction works better; and for corpora with higher comparability (e.g., the MSR corpus), the configuration of using both LCS and word alignments and the chunk-based fragment extraction outperforms the others.

---

[15]http://opennlp.sourceforge.net/

[16]http://research.microsoft.com/en-us/downloads/607d14d9-20cd-47e3-85bc-a2f65cd28042/

[17]The corpus is freely available at http://www.coli.uni-saarland.de/~rwang/pubs/ AMT2010Data.zip.

Configurations			
Aligner+ Phrase Extraction	LCS+ Chunk	Word+ N-Gram	LCS+Word+ Chunk
Our Corpus			
PARAPHRASE	15%	36%	32%
RELATED	21%	26%	21%
SUM	36%	**62%**	53%
The MSR Corpus			
PARAPHRASE	38%	44%	49%
RELATED	20%	19%	18%
SUM	58%	63%	**67%**

Table 5.7: Distribution of the extracted fragment pairs of our corpus and MSR corpus.

5.6.4 Discussion

Table 5.8 shows some examples from the best two settings. From our corpus, both simple paraphrases ("Governor ... said" and "Gov. ... announced") and more varied ones ("rose to fame as" and "the highlight of his career") can be extracted. It is clear that the smoothing and extraction algorithms do help with finding non-trivial paraphrases (shown in Figure 5.2). The extracted phrase "campaign was" shows the disadvantage of n-gram-based phrase extraction method, since the boundary of the fragment is improper. Using a chunker can effectively exclude such problems, as shown in the lower part of the table, where all the extracted paraphrases are grammatical phrases. Even from a parallel paraphrase corpus at the sentence level, the acquired fragment pairs (w/o context) may be non-paraphrases. For instance, the second pair from the MSR corpus shows that one news agency gives more detailed information about the launching site than the other, and the last example is also debatable, whether it's "under \$200" or "around \$200" depending on the reliability of the information source.

As far as we know, Munteanu and Marcu (2006)'s bilingual fragment extraction method has not yet been applied to the task of monolingual paraphrase extraction. Zhao et al. (2008) extracted paraphrase fragment

From Our Corpus: word aligner + n-gram-based phrase	
In San Juan, **Puerto Rico, Governor Pedro Rosello said** the storm could hit the US territory by Friday, ... In **Puerto Rico, Gov. Pedro Rossello announced** that banks will be open only until 11 a.m. Friday and ...	Paraphrase
Kunstler **rose to fame as** the lead attorney for ... **The highlight of his career** came when he defended ...	Paraphrase
... initiated **the air attacks** in response to Serb shelling of Sarajevo that killed 38 people Monday. The **campaign was** to respond to a shelling of Sarajevo Monday that killed 38 people.	Invalid

From MSR Corpus: LCS + word aligner + chunk-based phrase	
... Jordan Green, **declined to comment**. ... the prelate's private lawyer, said he **had no comment**.	Paraphrase
... to blast off between next Wednesday and Friday **from a launching site** in the Gobi Desert. ... to blast off as early as tomorrow or as late as Friday **from the Jiuquan launching site** in the Gobi Desert.	Related
... Super Wireless Media Router, which will be available in the first quarter of 2004, **at under $200**. The router will be available in the first quarter of 2004 and will cost **around $200**, the company said.	Related

Table 5.8: Some examples of the extracted paraphrase fragment pairs.

pairs from bilingual parallel corpora, and their log-linar model outperforms Bannard and Callison-Burch (2005)'s maximum likelihood estimation method with 67% to 60%. Notice that our starting corpora are (noisy) comparable corpora instead of parallel ones, and the approach is almost unsupervised, so that it can be easily scaled up to other larger corpora, e.g., news websites.

In sum, we presented our work on paraphrase fragment pair extraction from monolingual comparable corpora, inspired by Munteanu and Marcu (2006)'s bilingual method. We evaluated our intermediate results at each of the stages using MTurk. Both the quality and the quantity of the collected paraphrase fragment pairs are promising given the minimal supervision. As for the ongoing work, we are currently expanding our extraction process to the whole GIGAWORD corpus, and we plan to apply it to other comparable corpora as well.

For future work, we consider incorporating more linguistic constraints, e.g., using a syntactic parser (Callison-Burch, 2008), to further improve the quality of the collection. More importantly, applying the collected paraphrase fragment pairs to other NLP applications will give us a better view of the utility of this resource. As Bosma and Callison-Burch (2007) have utilized similar resources in the RTE task, a better way of using such paraphrase fragment pairs is still under exploration.

5.7 Summary

To sum up, we identify important issues encountered in using inference rules for textual entailment and propose methods to solve them. We explore the possibility of combining a collection obtained in a statistical and unsupervised manner, DIRT, with a hand-crafted lexical resource in order to increase the usefulness of inference rules for applications. We also investigate ways of effectively applying these rules. The experimental results show that although coverage is still not satisfactory, results in terms of precision are promising. Therefore, our method has the potential to be successfully integrated in the extensible framework described in Chapter 3 as a specialized module.

The error analysis points out several possible future directions. The tree skeleton representation we use needs to be enhanced in order to capture the relevant fragments of the text more accurately. A different issue remains the fact that a lot of rules we need for textual entailment

detection are still missing. Further investigations into the limitations of the DH as well as a classification of the knowledge we want to encode into the inference rules would be a step forward towards solving this problem.

At last, we present a pilot study of acquiring paraphrased fragment pairs using monolingual comparable corpora, which can be viewed as an alternative to the DIRT-style rules acquired based on the DH.

Part B: Extrinsic Approaches

6 Generalized Textual Semantic Relations

This chapter gives an overview of the following three chapters. Instead of tackling the standalone RTE task (as in Chapter 3, Chapter 4, and Chapter 5), we generalize the problem to include other relations other than entailment. We motivate this by going back to the scenario of an information seeker and showing the connection between entailment and other relations. We call these specific relations we consider *Textual Semantic Relations* (TSRs). We present a framework for handling all these relations simultaneously. Therefore, not only can entailment recognition benefit from other TSR recognition tasks, but also multiple tasks can be dealt with in one unified framework.

6.1   Motivation of the Approaches

As we are only interested in the cases where **T** is true, the intrinsic properties of entailment are *relevance* and *necessity* (Anderson and N.D. Belnap, 1975, Anderson et al., 1992). In (relevance) logic, *relevance* is defined based on the logic formulae by sharing atomic formulae (i.e., variables and constants) between premises and the conclusion; in the information seeking process, we observe similar phenomena. The information retrieved can also be validated by these two properties.

As we have mentioned in the introduction (Chapter 1), an information seeker usually retrieves information entailed by the exact goal what he or she wants to find. However, most of the information found is just something *related* to the goal[1]. Figure 6.1 roughly visualizes the relationship.
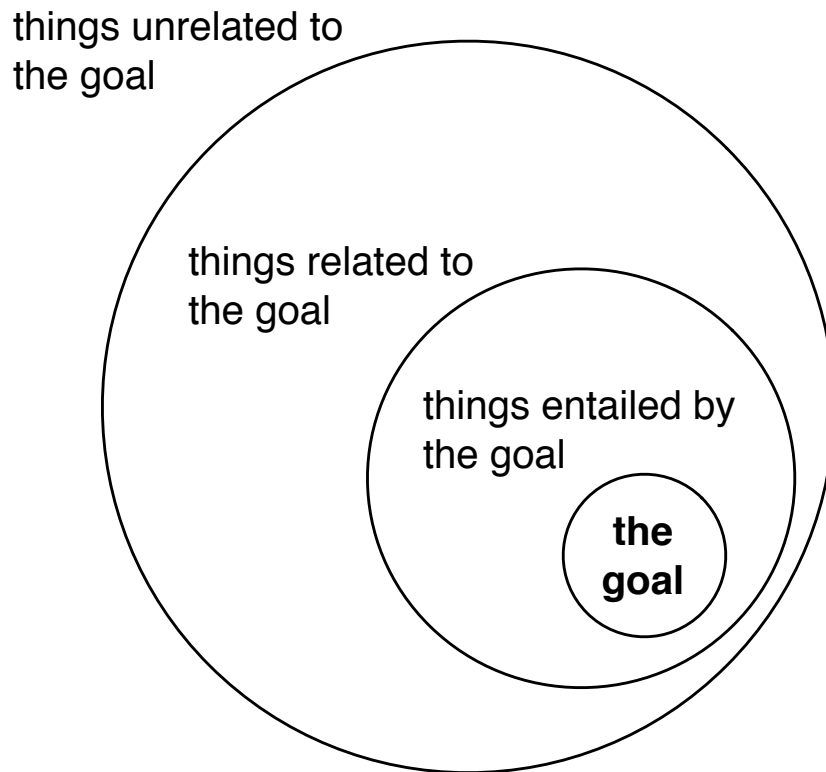
Figure 6.1: Things found by the information seeker

Comparing the things related to the goal with those entailed by the goal, we lose one property, which is *necessity*, but the *relevance* remains. Naturally, in order to be verified as an entailment relation, it has to pass the *relevance* test at the first place. That is exactly what many

---

[1]And the approach should be appropriate. Otherwise, most of the information is unrelated.

state-of-the-art RTE approaches do.

As we have seen in Section 2.3.3, a large group of RTE methods are based on overlapping information or similarity functions between **T** and **H**. They over-cover the entailment cases, and include non-entailment but *related* cases as well. If we take the standard three-way annotation used in the RTE community, Entailment, Contradiction, and Unknown, the risk is to bring Contradiction in.

Let us take the following **T**-**H** pair from the RTE-4 test set (Giampiccolo et al., 2009) as an example,

T: *At least five people have been killed in a head-on train collision in north-eastern France, while others are still trapped in the wreckage. All the victims are* **adults***.*

H: *A French train crash killed* **children***.*

This is a pair of two contradictory texts, where the events mentioned (i.e., "the train crash") in both **T** and **H** are assumed to refer to the same event[2]. The only contradictory part lies in the second sentence of **T** against **H**, that is, whether there are "children" among the "victims". Although the overlap information is quite large, it should still be annotated as Contradiction. On the other hand, in order to capture the contradictory part, we need to discover the related part at first.

In short, both Entailment and Contradiction have related **T** and **H**, and they are different from the unrelated text pairs. Figure 6.2 shows the relationship between the three annotations from the RTE challenges.

For the recognition task, many people directly do the three-way classification with selective features (e.g., Agichtein et al. (2009)) or different inference rules to identify entailment and contradiction simultaneously (e.g., Clark and Harrison (2009b)); while other researchers also extend their two-way classification system into three-way by performing a second-stage classification afterwards. We treat *relatedness* as a separate dimension from *consistency*. Given a related **T**-**H** pair, we further decide whether it is Entailment or Contradiction; if it is unrelated, it will be classified as Unknown. Note that in fact these relations cannot cover the whole area (e.g., the directionality of the entailment relation is ignored here), this is just a simplified figure to show that there are at least these two dimensions.

---

[2]See more details about the annotation guideline at `http://www.nist.gov/tac/tracks/2008/rte/rte.08.guidelines.html`
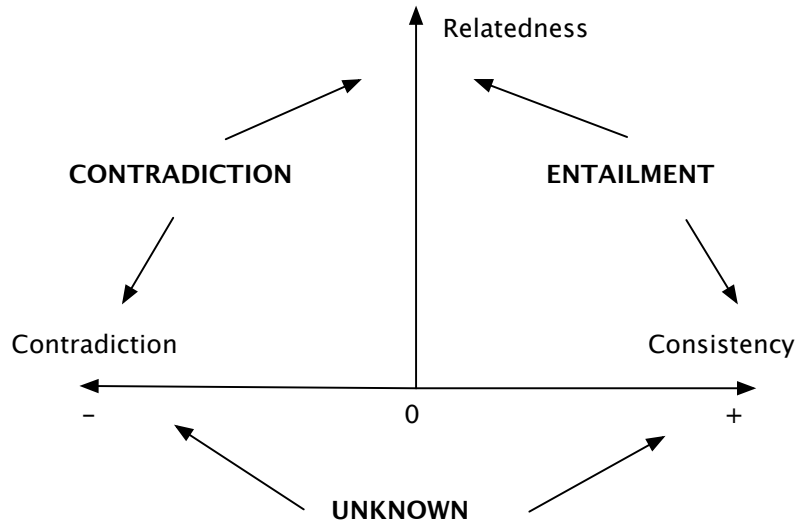
Figure 6.2: The relationship between the three relations

Strictly speaking, *related* is not a semantic relation between two texts. In logic, a semantic relation holds when it is true in all possible worlds; however *related* is highly dependent on the *user* and the context. For instance, chocolate is unrelated to an apple, if a fruit is required; but it is related when some food is wanted. Nevertheless, in practice, *relatedness* can help with semantic relation recognition by excluding other possibilities, i.e., pruning the search space.

Furthermore, we consider the other extreme of *relatedness*, where two texts are the same as or highly similar to each other. Semantically this becomes *equivalence* and textually *paraphrase*. In Figure 6.1, if we are at this extreme, we have already reached the exact goal. Now the question is what the other possible relations are, between two propositions, A and B:

- A is contradictory to B (two separate circles).

- A entails B (B is within A) or the other way around.

- A is equivalent to B (they are fully overlapping).

- A and B are overlapping, but not fully (e.g., being red and being an apple).

In addition, sometimes the relation between A and B is uncertain from the given context and it can be either of the four relations. Figure 6.3 visually shows the possibilities:
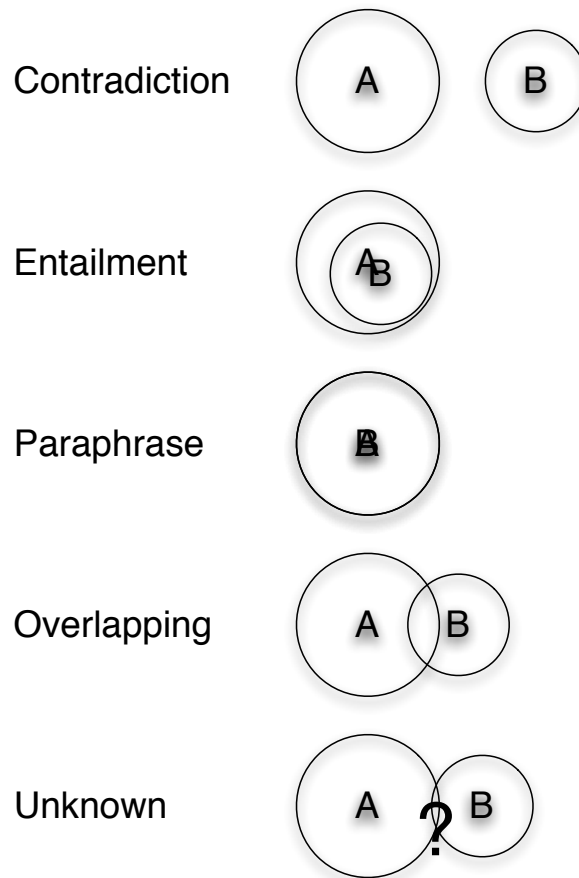
Figure 6.3: Possible semantic relations between A and B

If in all possible worlds, the first four are true, the relation holds in the actual world as well; however, for the last one, it can be one of the four relations above in some possible worlds (e.g., in one world, all the apples are red) or in our actual world. As we mentioned in Chapter 1, in practice, we only consider the world given by the context of the text pairs, and it is treated as our actual world (if we take commonsense knowledge into consideration).

Traditionally, the term *semantic relation* refers to relations that hold between the meanings of two words, e.g., synonymy, hypernymy, etc. These relations are usually situation-independent. However, the term has also been used in a wider sense to refer to relations between two linguistic expressions or texts, such as paraphrasing, textual entailment, and so on (Murakami et al., 2009). We call such relations **Textual** *Semantic Relations (TSRs)* in this dissertation to avoid confusion.

The task is also updated. Figure 6.4 shows the extension of the original
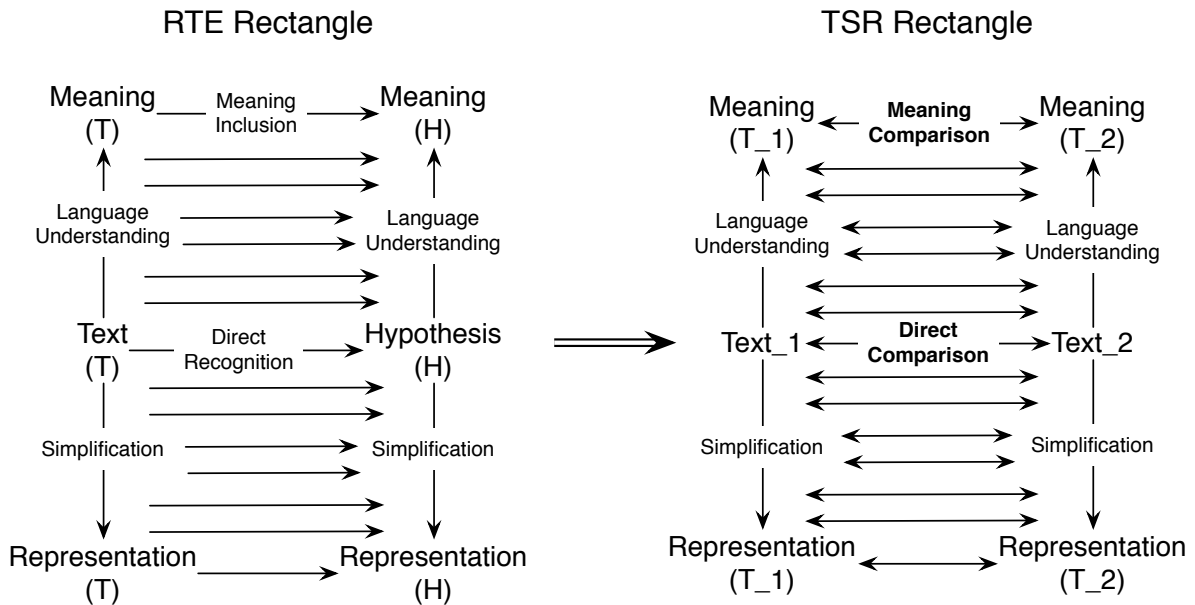
Figure 6.4: Comparison of the TSR rectangle and the RTE rectangle.

RTE rectangle into the TSR rectangle, although the problems remain the same:

- What is a proper meaning representation?

- How to detect all these relations?

6.2   The Framework

It may appear that we are making a hard problem even harder. In fact, we aim to handle the original problem (i.e., RTE) better by dealing with the harder one (i.e., TSR). When we do the corpus construction (Chapter 7.2), we have a more detailed version of six TSRs (Section 7.2.1), but the main task focuses on four relations, CONTRADICTION (C), ENTAILMENT (E), PARAPHRASE (P), and UNKNOWN (U). The OVERLAPPING cases are grouped into the UNKNOWN relation due to the less interest, and thus, UNKNOWN represents all the other cases apart from the first three relations.

In order to recognize or classify these relations, we need some measurements to better characterize them. *Relatedness* is one option. Although it is not a semantic relation (i.e., usually not true for all possible worlds), once we fix the context, we can use the *linguistically-indicated* relatedness as an approximation. It can be viewed as a weaker concept than

similarity but stronger than co-occurrence. For example, for CONTRA-DICTION, two texts are highly related, as shown in the example in the previous section ("adults" vs. "children"); for ENTAILMENT, this is also true, e.g., "apple" and "fruit"; PARAPHRASE has the highest relatedness, meaning they are (almost) the same; and for UNKNOWN it is possible to have either high or low relatedness. If two texts belong to any category, but do not have the above relations, like "apple" and "pear", the relatedness is quite high; while if they are totally different, e.g., "apple" and "car", they are unlikely to be related.

Apart from *relatedness*, we can further check how consistent the two texts are, how much information each has, and so on. If we go back to Figure 6.3, how much overlap they have is a good indicator for consistency. For CONTRADICTION, A and B have no overlap, while for PARAPHRASE they are fully overlapping. How much difference A and B have is another interesting measurement. Considering ENTAILMENT and PARAPHRASE, the sizes of A and B are different for the former, but the same for the latter.

In sum, we can define some measurements or latent features (partially) shared by the TSRs to differentiate them. We assume there exists a simplified low-dimension semantic relation space. Although the identification of effective dimensions is a complex question, we start with the following three ones:

- *Relatedness*

- *Inconsistency*

- *Inequality*

Different TSRs can be scattered in this space with different probability distributions.

Relatedness captures how relevant the two texts are. PARAPHRASE is one extreme (fully related), and some of the UNKNOWN cases are the other extreme, and the other cases stand in the middle. As we mentioned before, although it is not a semantic relation, excluding (some of) the UNKNOWN cases is still helpful to recognize other TSRs.

Inconsistency measures whether or how contradictory the two texts are. CONTRADICTION has the highest inconsistency, while ENTAILMENT and PARAPHRASE are not inconsistent at all. de Marneffe et al. (2008)

has already shown the importance of finding contradictions in many applications. Here, again, we hope to reduce the search space for the whole task.

Inequality mainly differentiates the asymmetric Entailment from the symmetric Paraphrase. Although the other two relations are symmetric as well, based on the unequal information contained in $\mathbf{T}$ and $\mathbf{H}$, we assume they are asymmetric.

All three features are numerical. In practice, *relatedness* is recall-oriented measurement, but the other two are precision-oriented. Notice that there are two approximations here:

1. The number of dimensions in the real semantic relation space is much larger.

2. The three dimensions we pick are not really orthogonal to each other (as shown in experiments in Section 9.3).

Nevertheless, we hope to benefit from the generality of these measures in the TSR recognition task. The empirical results are shown later. In particular, Chapter 8 shows the effectiveness of *relatedness* for the RTE task, and Chapter 9 shows the results for the whole TSR recognition task. In addition, a large benefit of such generalization is the wider range of available corpora resources, which is the focus of the coming Chapter 7.

6.3   Summary

In this chapter, we give an overview of our extrinsic approaches to deal with RTE. Instead of solving the problem in a standalone manner, we explore its connections to other tasks. We show in a broad view, different possible TSRs between two texts as well as the relationships between them. A unified framework is presented to handle all these TSRs simultaneously. We propose three measurements to characterize the (dis)similarity of the TSRs, *relatedness*, *inconsistency*, and *inequality*.

For future extensions, definitely more measurements need to be taken into consideration. The three-dimensional semantic space is over-simplified, although it is not trivial to achieve a better model. In a larger framework, we could also consider introducing the RTE architecture (i.e., specialized

modules) into other TSR recognition tasks. Chapter 10 expands on some of these issues.

In the following chapters we look at relatedness recognition (Chapter 8) and TSR recognition (Chapter 9), but firstly we present our work on corpora construction and discuss the other corpora used in our experiments.

7 Corpora Construction

This chapter[1] describes the corpora we used in this dissertation. We firstly give an overview of all the datasets we have, followed by a discussion about the methodologies of constructing them. Then we elaborate on two corpora we have constructed: one has a new annotation scheme of six categories of textual semantic relations with manual annotations; and the other uses the crowd-sourcing technique to collect the data from the Web. The final section provides a summary of this chapter.

---

[1]Section 7.2 has been published in (Wang and Sporleder, 2010), and it was a collaboration with Dr. Caroline Sporleder. Section 7.3 has been published in (Wang and Callison-Burch, 2010), and it was a collaboration with Prof. Dr. Chris Callison-Burch, who helped me to set up the tasks.

7.1 Existing Corpora

Resource building is the key step for many NLP tasks. Annotated datasets are especially important for system development. For instance, in the parsing community, almost all research on statistical parsing models of the previous decade relies on the Wall Street Journal (WSJ) sections of the Penn Treebank (PTB). A bilingual parallel corpus containing millions of sentence pairs makes a normal training size to build a statistical machine translation system.

In order to thoroughly understand all types of entailment, the currently available corpora are not satisfactory. Even being restricted to one specific task (i.e. binary classification of entailment vs. non-entailment), the size of the existing corpora is not large enough. Furthermore, the methods used to construct the corpora may lead to "artificial" datasets, whose distribution does not reflect the naturally occurring data. Although the yearly RTE challenge provides thousands of annotated text pairs, they are still far from *representative*.

As we mentioned in Section 3.1, many language phenomena are involved in the RTE task, which makes the limited datasets an even more serious problem. In particular, there are two issues involved here:

1. The annotation scheme of the corpus;

2. The methodology of data collection.

The annotation scheme of most datasets is a binary classification of one particular relation vs. the rest. For instance, ENTAILMENT vs. non-entailment, PARAPHRASE vs. non-paraphrase, and so on. From the RTE-3 pilot task to the RTE-5 challenge, the annotation was extended into ternary, ENTAILMENT, CONTRADICTION, and UNKNOWN. However, it is still quite unclear what exactly the UNKNOWN relation is.

The way of collecting the data also has an impact on the resulting corpus. For instance, Burger and Ferro (2005)'s automatic acquisition of positive entailment examples from news articles and their headlines may lead to an RTE corpus similar to the *summarization* task, although the latter can be viewed as one particular case of entailment.

Before we explore these two issues for each corpus, we firstly give an overview of both existing corpora and newly constructed ones (Table 7.1). The numbers below denote the number of **T-H** pairs contained in each set.

RTE-2&3 (3200)	Entailment (1578)		Non-Entailment (1622)		
RTE-4&5 (2200)	ENTAILMENT (1100)		CONTRADICTION (330)	UNKNOWN (770)	
PETE (367)		YES (194)	NO (173)		
MSR (5841)	Paraphrase (3940)	Non-Paraphrase (1901)			
TSR (260)	Equality (3)	F/B Entailment (10/27)	Contradiction (17)	Overlapping (72)	Independent (131)
AMT (584)		Facts (406)	Counter-Facts (178)		

Table 7.1: Annotation scheme comparison of the different corpora.

We briefly introduce the RTE, PETE, and MSR corpora in the rest of this section and leave the other two, the TSR corpus and the AMT corpus, which are constructed by ourselves, for the next two sections. Notice that the five corpora discussed here do not cover all the existing datasets. We have already mentioned many other available resources in Section 2.1.1. Here we just focus on these five, because we use them for our evaluation presented in Chapter 9.

7.1.1 The RTE Corpora

The RTE Corpora are a combination of RTE-2 (1600 **T-H** pairs) (Bar-Haim et al., 2006), RTE-3 (1600 **T-H** pairs) (Giampiccolo et al., 2007), RTE-4 (1000 **T-H** pairs) (Giampiccolo et al., 2009), and RTE-5 (1200 **T-H** pairs) (Bentivogli et al., 2009) datasets[2]. The former two have the original two-way annotation, ENTAILMENT and non-entailment; and in the latter two, a third category was added, resulting in ENTAILMENT, CONTRADICTION, and UNKNOWN[3]. Notice that the ENTAILMENT cases here actually include PARAPHRASE as well, which can be viewed as a bi-directional entailment. UNKNOWN also contains many other cases.

Table 7.2 shows some examples. The two-way judgement is based on the following four criteria:

1. As entailment is a directional relation, the hypothesis must be entailed by the given text, but the text need not be entailed by the hypothesis.

---

[2]http://www.nist.gov/tac/data/RTE/index.html
[3]We did not include the unofficial three-way annotation of the RTE-3 pilot task.

Source	Task	Text	Answer
RTE-3	IE	**T**: At the same time the Italian digital rights group, Electronic Frontiers Italy, has asked the nation's government to investigate Sony over its use of anti-piracy software.	NO
		H: Italy's government investigates Sony.	
RTE-3	QA	**T**: Aeschylus is often called the father of Greek tragedy; he wrote the earliest complete plays which survive from ancient Greece. He is known to have written more than 90 plays, though only seven survive. The most famous of these are the trilogy known as Orestia. Also well- known are The Persians and Prometheus Bound.	YES
		H: "The Persians" was written by Aeschylus.	

Table 7.2: Examples of the RTE corpora (with two-way annotations)

2. The hypothesis must be fully entailed by the text. Judgment must be NO if the hypothesis includes parts that cannot be inferred from the text.

3. Cases in which inference is very probable (but not completely certain) were judged as YES.

4. Common sense world knowledge was assumed, e.g., the capital of a country is situated in that country, the prime minister of a state is also a citizen of that state, and so on.

Although the RTE-2 and RTE-3 datasets are balanced for ENTAILMENT (or YES) and non-entailment (or NO), the distribution in the real data is unlikely to be the same. There are many cases of non-entailment, a random pair of text, a contradictory pair of text, **H** is only partially entailed, and so on. Consequently, RTE-4 and RTE-5 take CONTRADICTION out of the non-entailment pool and call the rest UNKNOWN. The criteria are:

- **T** entailed **H** - in which case the pair was marked as ENTAILMENT.

- **T** contradicted **H** - in which case the pair was marked as CONTRA-
  DICTION.

- The truth of **H** could not be determined on the basis of **T** - in which
  case the pair was marked as UNKNOWN.

Source	Task	Text	Answer
RTE-4	IR	**T**: The Dalai Lama today called for Tibetans to end protests against the Beijing Olympics, also telling MPs in London he would happily accept an invitation to attend the event if relations with China improved. **H**: China hosts Olympic games.	E
RTE-4	SUM	**T**: Kingdom flag carrier British Airways (BA) has entered into merger talks with Spanish airline Iberia Lineas Aereas de Espana SA. BA is already Europe's third-largest airline. **H**: The Spanish airline Iberia Lineas Aereas de Espana SA is Europe's third-largest airline.	C
RTE-5	IE	**T**: Henan province has registered seven dead children and 4,761 HFMD cases. Shandong has reported five children dead from HFMD and 3,280 cases to deal with. HFMD can start from a variety of viruses of which Enterovirus 71 (EV-71) is the most common, followed by the Coxsackie A virus (Cox A16). There is an Incubation period from time of contact to appearance of symptoms between three to seven days. **H**: Shandong is not far from Henan province.	U

Table 7.3: Examples of the RTE corpora (with three-way annotations)

Table 7.3 shows some examples. The distribution of these three annotation labels in the dataset is 50% ENTAILMENT, 35% UNKNOWN, and 15% CONTRADICTION. Nevertheless, the problem of representative negative examples still remains. Previously, it was difficult to define what is a *non-entailment*; and currently, it is not trivial to find a good scope for UNKNOWN. In fact, instead of filtering out some cases at the first place (e.g., a random pair of texts), it is more natural to keep text pairs with different possible (semantic) relations in the corpus, i.e., to keep the gradient of similarity or relatedness. This is also one of the motivations to construct the TSR corpus, which will be introduced in the next section.

In addition, the RTE-5 data are different from the previous challenges in the following two aspects:

1. The **T**'s are longer, up to 100 words, whereas in RTE-4 the average length is about 40 words. Longer texts introduce discourse phenomena, such as coreference, which were not present in the previous data sets.

2. Texts taken from a variety of freely available sources to avoid copyright problems, and are not edited from their source documents. In this way, systems are asked to handle real text that may include typo-graphical errors and ungrammatical sentences.

Each pair of the dataset was judged by three annotators. Pairs on which the annotators disagreed were discarded. For instance, on the RTE-3 test set, the average agreement between each pair of annotators who shared at least 100 examples was 87.8%, with an average Kappa level of 0.75.

The data in the RTE corpora were semi-automatically obtained from four application scenarios, information extraction (IE), information retrieval (IR), question answering (QA), and multi-document summarization (SUM) (Bar-Haim et al., 2006, Giampiccolo et al., 2007, 2009, Bentivogli et al., 2009). These application scenarios can be described as follows:

IE IE was inspired by the Information Extraction (and Relation Extraction) application, where texts and structured templates were replaced by **T-H** pairs. Hypotheses were taken from the relations tested in the ACE tasks, while texts were extracted from the outputs of actual IE systems, which were fed with relevant news articles. Correctly

extracted instances were used to generate positive examples, and incorrect instances to generate negative examples. The same material was used and the news articles were also used to manually generate entailment pairs based on ACE relations[4], simulating the extraction process performed by IE systems. New relations, such as "X discover Y", "X win Y", etc., were produced both to be processed by IE systems and to manually generate **T-H** pairs from collected news articles;

IR In this setting, the hypotheses were propositional IR queries, e.g., "corn prices increase". Texts that did or did not entail the hypotheses were selected from documents retrieved by different search engines such as Google, Yahoo and MSN, for each hypothesis. In this application setting, the given propositional hypotheses are assumed to be entailed by relevant retrieved documents;

QA Both questions taken from the datasets of official QA competitions, such as TREC QA[5] and QA@CLEF datasets[6], and questions produced specifically for the purposes of RTE were fed to actual QA systems, which retrieved answers from the Web. Then, human annotators transformed the question-answer pairs into **T-H** pairs. An answer term of the expected answer type was picked from the answer passage - either a correct or an incorrect one. The question was turned into an affirmative sentence plugging in the answer term. **T-H** pairs were generated, using the affirmative sentences as hypotheses (**H**'s) and the original answer passages as texts (**T**s). Examples for which the entailment did not hold were created by producing **H**'s where the piece of information answering the implied question was not relevant or contradicted the content of the **T**;

SUM **T**'s and **H**'s were sentences taken from a news document cluster, a collection of news articles that describe the same news item. Annotators were given the output of multi-document summarization systems - including the document clusters and the summary generated for each cluster. Then they picked sentence pairs with high lexical

---

[4]ACE 2004 information extraction templates, from the National Institute of Standards and Technology (NIST). `http://www.itl.nist.gov/iad/mig//tests/ace/2004/`

[5]TREC IR queries and TREC-QA question collections, from the National Institute of Standards and Technology (NIST). `http://trec.nist.gov/`

[6]CLEF IR queries and CLEF-QA question collections, from DELOS Network of Excellence for Digital Libraries. `http://www.clef-campaign.org/`, `http://clef-qa.itc.it/`

overlap, preferably where at least one of the sentences was taken from the summary (this sentence usually played the role of **T**). For positive examples, the hypothesis was simplified by removing sentence parts, until it was fully entailed by **T**. Negative examples, where the entailment did not hold, were produced in a similar way, i.e., taking away parts of **T** so that the final information contained in **H** either contradicted the content of **T**, or was not enough to determine whether **T** entailed **H**.

RTE-2 and RTE-3 both used all four scenarios. Each scenario contributed equally to the final datasets. RTE-4 also made use of all the scenarios, but focused more on IE and IR, which were assumed to be more difficult than the other two. IE and IR both had 300 text pairs, and QA and SUM had 200 pairs. RTE-5 excluded SUM and had the same number of text pairs for the other three scenarios. In addition, all challenges except RTE-4 had an equal size of development and test set. RTE-4 only had a test set.

These four scenarios do not necessarily cover all types of entailment. Therefore, the data collected by the RTE challenges focus more on NLP tasks rather than linguistic phenomena[7], and the semi-automatic construction method may also lead to artificial sentences instead of naturally-occurring utterances. We take this issue into account when constructing the TSR corpus and the AMT corpus. The texts (both **T** and **H**) of the former corpus were all extracted from news articles (Section 7.2); and the hypotheses of the latter corpus were proposed by non-expert annotators without much linguistic or NLP knowledge (Section 7.3).

7.1.2   The PETE Corpus

The PETE Corpus is taken from the SemEval-2010 Task #12, Parser Evaluation using Textual Entailment[8] (Yuret et al., 2010). The dataset contains 367 pairs of texts in all and focuses on entailments involving mainly syntactic information. The annotation is two-way, YES means ENTAILMENT and NO means non-entailment. Each hypothesis only concerns one syntactic phenomenon. Therefore, the entailment relation is directional, excluding the paraphrases. Table 7.4 shows some examples.

---

[7]Compared with the FraCaS dataset (Cooper et al., 1996).
[8]`http://pete.yuret.com/guide`

Text	Answer
T: Any lingering **suspicion** that this **was a trick** Al Budd had thought up **was dispelled**.	
H_1: The suspicion was dispelled.	YES
H_2: The suspicion was a trick.	NO

Table 7.4: Examples of the PETE corpus

The way of constructing hypotheses is also semi-automatic. It contains three main steps:

1. Identify syntactic dependencies that are challenging to state of the art parsers;

2. Construct short entailment sentences that paraphrase those dependencies;

3. Identify the subset of the entailments with high inter annotator agreement

In particular, the entailments were built around two content words that are syntactically related. When the two content words were not sufficient to construct a grammatical sentence, one of the following techniques was used:

- Complete the two mandatory elements using the words "somebody" or "something", e.g., to replace "John kissed Mary." by "John kissed somebody."

- Make a passive sentence to avoid using a spurious subject, e.g., to replace "John kissed Mary." by "Mary was kissed."

- Make a copular sentence to express noun modification, e.g., to replace "The big red boat sank." by "The boat was big."

Each entailment was then tagged by five untrained annotators. The results from the annotators whose agreement with the gold parse fell below 70% were eliminated. The entailments for which there was unanimous agreement of at least three annotators were kept. The instructions for the annotators were brief and targeted people with no linguistic background. They chose to rely on untrained annotators on a natural inference task rather than trained annotators on an artificial tagging task,

which is consistent with our idea when we construct the AMT corpus (Section 7.3). The whole idea of building an entailment corpus focusing on single syntactic phenomena is also consistent with our extensible architecture consisting of specialized RTE modules presented in Chapter 3.

7.1.3 The MSR Corpus

Text	Answer
T_1: Amrozi accused his brother, whom he called "the witness", of deliberately distorting his evidence.	YES
T_2: Referring to him as only "the witness", Amrozi accused his brother of deliberately distorting his evidence.	
T_1: Yucaipa owned Dominick's before selling the chain to Safeway in 1998 for $2.5 billion.	NO
T_2: Yucaipa bought Dominick's in 1995 for $693 million and sold it to Safeway for $1.8 billion in 1998.	

Table 7.5: Examples of the MSR corpus

The MSR Corpus[9] is a paraphrase corpus provided by Microsoft Research (Dolan and Brockett, 2005). It is a collection of manually annotated sentential paraphrases. This dataset consists of 5841 pairs of sentences which have been extracted from news sources on the web, along with human annotations indicating whether each pair captures a paraphrase or a semantic equivalence relationship. Table 7.5 shows two examples.

The annotated sentence pairs were randomly selected from 20,574 candidate pairs, which were filtered by an SVM-based classifier. These candidate paraphrase pairs were examined by two independent human judges. Each judge was asked whether the two sentences could be considered *semantically equivalent.* Disagreements were resolved by a third judge, with the final binary judgment reflecting the majority vote. Consequently, the final annotation is binary, PARAPHRASE or non-paraphrase.

---

[9]`http://research.microsoft.com/en-us/downloads/607d14d9-20cd-47e3-85bc-a2f65cd28042/`

The original candidate sentence pairs were distilled from a database of 13,127,938 sentence pairs, extracted from 9,516,684 sentences in 32,408 news clusters collected from the World Wide Web over a 2-year period. The methods and assumptions used in building this initial data set are discussed in (Quirk et al., 2004) and (Dolan et al., 2004). Heuristics based on shared lexical properties and sentence position in the document were employed to construct the initial database, and large number of sentence pairs were excluded whose differences might be attributable only to typographical errors, variance between British and American spellings, and minor editorial variations.

The annotation of this corpus does not directly correlate to the entailment relation. However, there are at least three relevant issues:

1. PARAPHRASE can be viewed as a bi-directional ENTAILMENT, which means the positive examples in the MSR corpus are certainly positive examples of ENTAILMENT;

2. The original RTE corpus may also include PARAPHRASE pairs, since whether **H** entails **T** is neither required nor banned;

3. Non-paraphrase may be potentially a good source for ENTAILMENT, including both positive and negative cases.

In addition, the size of this corpus is also relatively large. Therefore, it is also included in our dataset for evaluation.

7.2 The TSR Corpus

Although *entailment* is a semantic relation, RTE is usually beyond that level. The task is defined to discover the relation between two texts, which usually contain more than one sentence. Most previous research on RTE focuses on the lexical, syntactic, and semantic levels. Studies that have looked at the discourse level have been typically restricted to a specific discourse context, for example, whether examples of entailment can be acquired from news texts and their corresponding headlines (Burger and Ferro, 2005).

So far, there has been little work that has investigated the connection between discourse and textual semantic relations (TSRs), such as the relation between CAUSE and ENTAILMENT, or CONTRAST and CONTRADICTION, etc. In general, (strict) entailment or repetition is unlikely to

appear frequently in a naturally occurring discourse, since redundant information content violates the Gricean maxim of manner (Grice, 1975). Nonetheless, there are situations in which information is at least partly repeated, e.g., in restatements or summaries.

Consequently, we have constructed a corpus on textual semantic relations based on existing discourse treebanks by analyzing the relationship between discourse relations and semantic relations.

Two research issues are addressed here:

1. An alternative way of constructing a (balanced) corpus for RTE or TSR recognition;

2. A better understanding of discourse and semantic relations.

Below, we present our work on constructing the corpus by making use of an existing treebank annotated with discourse relations. We extract adjacent text span pairs and group them into six categories according to the different discourse relations between them. After that, we present the details of our annotation scheme (Section 7.2.1), which includes six textual semantic relations, *backward entailment*, *forward entailment*, *equality*, *contradiction*, *overlapping*, and *independent*. We also discuss some ambiguous examples to show the difficulty of such annotation tasks, which cannot be done easily by an automatic mapping between discourse relations and semantic relations (Section 7.2.2). We have two annotators and each of them performs the task twice. The basic statistics on the constructed corpus look promising: we achieve 81.17% of agreement on the six semantic relation annotation with a .718 kappa score, and it increases to 91.21% with a .775 kappa score if we collapse the last two labels (Section 7.2.3).

7.2.1   Annotation Scheme and Results

To obtain data annotated with discourse relations, we used the RST Discourse Treebank (RST-DT)[10]. RST defines a set of 24-30 relatively fine-grained discourse relations, such as CONTRAST, RESTATEMENT, ELABORATION or BACKGROUND. Most relations are binary and link a *nucleus* (N) (i.e., a more important text span) to a *satellite* (S) (i.e., the less important span). We extracted all relations holding between adjacent sen-

---

[10]Available from the LDC: `http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2002T07`

tences from the RST-DT, thus excluding relations between sub-sentential clauses or larger pieces of text.

By looking at the discourse relations mentioned above, we can already observe some potentially relevant TSRs. For instance, if two adjacent texts have the RESTATEMENT relation, they may be a (non-strict) paraphrase to each other. An ELABORATION relation can exist between two texts, where a backward entailment may also hold, e.g., a concrete story entails a short headline. A CONTRAST relation may contain a contradiction between two texts, although people usually do not make totally contradictory utterances. In the most common situation, when the two text spans have no such strong TSRs (e.g., the BACKGROUND relation), we assume that they are still relevant to each other in some sense. They may mention the same entity, different steps of one procedure, consequent actions, etc.

Since the inventory of RST relations is relatively fine-grained, we manually grouped the relations into six classes, more or less following the "relation groups" in the RST-DT annotation manual[11]. Each group contains related discourse relations and we hypothesize that relations within a given group also behave similar with respect to the TSRs to which they can be mapped. The resulting six relation groups are:

- background : BACKGROUND, CIRCUMSTANCE;

- elaboration : ELABORATION-SET-MEMBER, ELABORATION-PROCESS-STEP, ELABORATION-OBJECT-ATTRIBUTE, ELABORATION-GENERAL-SPECIFIC, EXAMPLE, ELABORATION-ADDITIONAL;

- explanation : EXPLANATION-ARGUMENTATIVE, EVIDENCE, PURPOSE, REASON;

- consequence : CONSEQUENCEN, CONSEQUENCES, CONSEQUENCE, CAUSE, CAUSE-RESULT, RESULT;

- contrast : ANTITHESIS, CONCESSION, CONTRAST, INTERPRETATIONS, INTERPRETATIONN;

- restatement : RESTATEMENT, SUMMARYS, SUMMARYN.

We excluded ENABLEMENT, which is grouped with PURPOSE in the RST-DT manual, because the nucleus in ENABLEMENT is supposed to be unrealized. We also excluded EVALUATION, which is grouped with

---

[11]http://www.isi.edu/~marcu/software/manual.ps.gz

INTERPRETATION, because unlike INTERPRETATION, both text spans of EVALUATION are "attributed to the same agent", i.e., there is no contrastive aspect. The rest of the excluded relations, e.g., LIST, SEQUENCE, etc., were disregarded due to two reasons: 1) we hypothesize that these relations are not interesting for semantic relations, especially for the entailment relation; and 2) some of them occur very infrequently in the corpus, making it impossible to make any empirical statements about them.

The extracted RST-DT examples were then manually labelled with TSRs. We define eight annotation labels:

- FE - Forward Entailment : There is an entailment relation between the two text spans, and the direction is from the first one to the second one.

- BE - Backward Entailment : There is an entailment relation between the two spans, and the direction is from the second one to the first one, e.g., Example 4 in Table 7.9.

- E - Equality : The two spans are paraphrases of each other, or the entailment relation holds in both directions (forward and backward). The meaning is (almost) the same, like Example 12 in Table 7.12.

- C - Contradiction : There is a contradiction between the two spans. The meaning or information of (some parts of) the two spans are contradictory to each other. For instance, Example 6 in Table 7.9.

- O - Overlapping : None of the above relations holds, but the spans are relevant to each other and share much meaning or information, like Example 1 in Table 7.7.

- I - Independent : There are no overlapping events between two text spans, even though there is one entity mentioned in both, like Example 11 in Table 7.11.

- ? - Uncertain : The question mark can be combined with the first four labels, meaning that the relation holds not strictly, but loosely from the annotator's point of view. For instance, Example 5 in Table 7.9 is not a strict FE, but the information in the second span can be inferred from the first span with a relatively high probability.

- F - False : The example is not valid. It may be that the sentence extracted from the corpus is incomplete or hard to understand without further context.

Our goal was to capture the whole spectrum of different relations between meanings of two texts. On the dimension of overlapping information, we have little overlapping information (i.e. I), some overlapping (i.e. O), and fully the same (i.e. E); on the dimension of consistency, we have both contradictory relations (i.e. C) and consistent relations (i.e., all the other relations). In particular, we also incorporate the directionality of the ENTAILMENT relation (i.e. FE and BE) vs. the bi-directional PARAPHRASE, which has not been fully explored in the field yet.

The annotations were done by two experienced annotators. Annotating TSRs is a relatively hard task, particularly when it is done on naturally occurring examples, because, as was mentioned before, totally clear cases of entailment and contradiction are relatively rare compared to artificially constructed examples. To arrive at a reasonably reliable annotation, the annotation was done in two rounds. Initially, the annotators only labelled a subset of the data (100 examples). The annotators then discussed examples on which they disagreed with the aim of arriving at a more consistent annotation. The discussion phase also helped in making the annotation guidelines more precise. In the second round, the remaining examples were labelled. So far, we have annotated 319 text pairs, and among them there are 239 (75%) valid pairs[12], i.e., not labelled as F.

Annotations	strict		loose	
	Six-Way	Collapse I&O	Six-Way	Collapse I&O
Agreement	79.50%	89.54%	81.17%	91.21%
Kappa	.696	.736	.718	.775

Table 7.6: Inter-annotator agreement

To assess the reliability of our annotation, we computed the inter-annotator agreement (excluding instances labelled as F and ?). The results are shown in Table 7.6[13]. Under the 'strict' agreement evaluation scheme labels with and without a question mark (e.g., FE vs. FE?) were considered different, under a 'loose' evaluation scheme the question marks were disregarded. We also computed the agreement after collapsing the classes 'independent' (I) and 'overlap' (O), since these two classes

---

[12]Together with the first 100 test examples, we collect 260 valid pairs in all.

[13]The difference between *strict* and *loose* is that the latter ignores the question mark in the annotations. And "Collapse I&O" means we treat I and O as one relation.

were often shown to be difficult to distinguish[14] (see Section 7.2.2). The inter-annotator agreement is 79.5% for the strict six-way annotation and 91.2% for the loose five-way annotation. We also computed the Kappa statistic (Krippendorff, 1980), which corrects the percentage agreement for expected chance agreement. Our kappa scores range from .696 to .775 (see Table 7.6), which is considered as good agreement. Hence our annotations are generally fairly reliable.

In the next section, we provide some example annotations to make the definition of the TSRs more concrete and we also discuss some borderline cases.

7.2.2   Illustrative Examples

Relation Group: background			
Id	Sentences	#1	#2
1	**The engineering company** was acquired in a takeover earlier this year by **the giant Reliance textile group**.	O	O
	Although **Larsen & Toubro** hadn't raised money from the public in 38 years, **its new owners** frequently raise funds on the local market.		

Table 7.7: Examples of the annotated text pairs for the relation group: background

To illustrate our annotation scheme, we show some examples from our data in Table 7.7 - Table 7.12. Generally our annotators agreed well on assigning which TSR to a given text pair (see Section 7.2.1). However, some distinctions are difficult to make. In this section, we discuss those examples, for which the distinction between two labels is not straightforward.

An annotation decision that proved particularly difficult was the distinction between I and O. In practice, we use the number of shared entities as one criteria, namely, category I allows at most one shared entity between the two text spans, while examples with a higher overlap should

---

[14]For all the experiments on this corpus, we also collapse these two labels, since we view the other labels more reliable.

	Relation Group: elaboration		
Id	Sentences	#1	#2
2	The contract signing represented a major step in the long-planned petrochemical project.	I	BE
	At an estimated \$360 million, the project would represent the single largest foreign investment in the Philippines since President Corazon Aquino took office in February 1986.		
3	**Eli Lilly & Co.**, the Indianapolis-based drug manufacturer, dominates the U.S. human insulin market with its product known as Humulin.	I	O
	Lilly is building plants to make the insulin in Indianapolis and Fagershein, France.		

Table 7.8: Examples of the annotated text pairs for the relation group: elaboration

be labelled O (unless one of the other relations holds). A relatively clear case of I is Example 11 in Table 7.11, where there are no obvious shared entities between the two spans. In contrast, Example 1 in Table 7.7 is a fairly straightforward case of an overlap relation (O): "The engineering company" is co-referent with "Larsen & Toubro" and "the giant Reliance textile group" is co-referent with "its new owners".

Although the distinction is easy for most of the cases, there are still some tricky ones. For instance, in Example 7 (Table 7.9), both annotators agree that both spans evoke a reference to "sales" but one annotator is not sure whether "some traders" in the first span are the same as "platinum and palladium traders" in the second span. Example 10 (Table 7.11) is more interesting. "These goals" in the second span are generally referring to those mentioned in the proposal (from the first span), but it is unclear whether they should be treated as single entity or multiple entities.

Example 5 in Table 7.9 illustrates the use of a question mark in combination with one of the annotation labels. In this example, it is difficult to verify the quantifier "every" in the first text, but we still think the forward entailment relation holds, albeit loosely. As we mentioned at the beginning, it is almost impossible to find strict entailment or repetition in a naturally occurring discourse since it violates the Gricean maxim

	Relation Group: explanation		
Id	Sentences	#1	#2
4	Ford Motor Co. and Chrysler Corp. representatives criticized Mr. Tonkin's plan as unworkable.	BE	BE
	It "is going to sound neat to the dealer except when his 15-day car supply doesn't include the bright red one that the lady wants to buy and she goes up the street to buy one," a Chrysler spokesman said.		
5	Many of the problems you presented exist in every part of this country.	FE?	FE?
	Poverty is only two blocks from President Bush's residence.		
6	In response to questions after the annual meeting, Mr. Miller said the company is no longer looking for an equity investor.	C	C
	During the summer, Wang executives had said they might seek outside investment.		
7	Some traders were thought to be waiting for **the auto sales** report, which will be released today.	O	I
	Such sales are watched closely by platinum and palladium traders because both metals are used in automobile catalytic converters.		

Table 7.9: Examples of the annotated text pairs for the relation group: explanation

of Manner. Instead one finds cases of 'soft entailment' where one span follows from the other with a reasonably high probability. Annotators sometimes differ with respect to how high they estimate this probability to be, and annotate FE or FE?, depending on their own interpretation of the likelihood of entailment.

Entailment relations can also be interpreted differently. The annotators agreed on the BE relation for Example 4 in Table 7.9, while Example 2 in Table 7.8 and Example 8 in Table 7.10 are not agreed on. In Example 2, one annotator considers that a big project does not necessarily

Relation Group: consequence			
Id	Sentences	#1	#2
8	Recession fears are springing up again among investors.	BE	I
	Analysts say that the selling of cyclical stocks yesterday will be followed by a sell-off in shares of companies with big debt loads on their balance sheets.		

Table 7.10: Examples of the annotated text pairs for the relation group: consequence

Relation Group: contrast			
Id	Sentences	#1	#2
9	Gulf Power said in May that an internal audit had disclosed that at least one vendor had used false invoices to fund political causes.	C	O
	But the company said the political contributions had been made more than five years ago.		
10	The proposal reiterates the U.S. desire to scrap or reduce a host of trade-distorting subsidies on farm products.	I	O
	But it would allow considerable flexibility in determining how and when **these goals** would be achieved.		
11	Rates are determined by the difference between the purchase price and face value.	I	I
	Thus, higher bidding narrows the investor's return while lower bidding widens it.		

Table 7.11: Examples of the annotated text pairs for the relation group: contrast

mean the contract signing is important (i.e., I), while the other annotator understands "big" as "financially significant", which does entail "important" (i.e., BE). In Example 8, one annotator thinks "the selling of cyclical stocks" does not entail "recession fears" (i.e., I), while the other annotator feels that "sell-off" gives an impression of such fears

Relation Group: restatement			
Id	Sentences	#1	#2
12	"Anne doesn't believe in blandness," said Ms. Smith. "She wants things to be exciting."	E	E

Table 7.12: Examples of the annotated text pairs for the relation group: restatement

(i.e., BE). In addition, these examples containing abstraction and inference can hardly be labeled as O, since shared (concrete) entities are difficult to find.

For contradiction cases, both annotators agree on Example 6 in Table 7.9, since there is a sharp contrast between what "Wang executives had said" in the summer and what they (i.e., "Mr. Miller") say now. However, they disagree on Example 9 in Table 7.11. One annotator interprets "in May" as an implicit mentioning of (May of) "this year", which is contradictory to "more than five years ago" in the other text; while the other annotator does not consider them comparable to each other, thus annotating O.

The examples above reveal some challenges of the annotation task. Although we achieved a relatively high inter-annotator agreement (Table 7.6), some annotations are still debatable. Nevertheless, after two-round discussion, we extracted the agreed text pairs and constructed the TSR corpus.

7.2.3   Corpus Statistics

In this section, we present some statistics of the TSR corpus.

For this study, we were particularly interested in whether specific discourse relations tend to correlate with particular TSRs. Table 7.13 provides some basic statistics of the corpus, as well as the distribution of the discourse relation groups versus the TSR annotations[15]. Note that we only count the agreed text pairs in this table.

It can be seen that I and O are the most frequent relations, holding between 50.52% and 28.84% of the text pairs, respectively. The other

---

[15]BG stands for BACKGROUND; CS for CONSEQUENCE; CT for CONTRAST; EL for ELABORATION; EX for EXPLANATION; and RE for RESTATEMENT.

	I	O	C	FE	BE	E	all	%
BG	18	12	0	0	0	0	30	15.46%
CS	8	6	0	3	1	0	18	9.28%
CT	22	11	13	1	2	0	49	25.26%
EL	29	17	1	0	4	1	52	26.80%
EX	21	7	1	1	9	0	39	20.10%
RE	0	1	0	1	2	2	6	3.09%
all	98	54	15	6	18	3	194	100.0%
%	50.52%	28.84%	7.73%	3.09%	9.28%	1.55%	100.0%	

Table 7.13: Distribution of the annotation labels across the relation groups

relations are comparably rare, especially true bi-directional entailment, i.e. equivalence (E), which only occurs three times. This is not surprising since we hypothesized that true entailment is rare in naturally occurring text. Backward entailment (BE) is more frequent than forward entailment (FE), and contradictions (C) are more or less equally frequent as backward entailments.

With respect to discourse relations, CONTRAST, ELABORATION, and EXPLANATION occur most often in our sample and these three relations are more or less equally frequent. While our sample is a bit biased with respect to discourse relations, since we excluded some relations, the fact that these three relations are relatively frequent between adjacent text spans is to be expected. RESTATEMENT is the least frequent relation, which is also expected.

Given the relatively small data set, it is difficult to make definite statements about the correlation of different discourse relations and TSRs, however, some trends are observable. First, it seems that TSRs distribute unevenly across different discourse relations. For instance, CONTRAST contains almost all the C cases (13/15), while ELABORATION and EXPLANATION have the most BE cases (4/18 and 9/18). As expected, RESTATEMENT relations tend to correlate with some form of entailment (E, BE, or FE), five out of six restatements involve entailment.

It is also interesting to look at the unobserved pairings of discourse relation and TSR. Some of these seem very plausible. For instance, one would not expect contradiction or independence for a RESTATEMENT relation. Likewise, one would not expect to find a bi-directional entailment for a CONTRAST relation.

However, while some trends are observable and intuitive, it is also clear from the data that there is no clear one-to-many or many-to-one mapping between discourse relations and TSRs. Most discourse relations can co-occur with most TSRs and vice versa. This suggests that discourse relations and TSRs capture different and partly independent aspects of meaning.

7.3   The AMT Corpus

As we mentioned in the overview (Section 7.1), the datasets used in the RTE challenges were collected by extracting paragraphs of news text and manually constructing hypotheses. For the data collected from information extraction task, the **H** is usually a statement about a relation between two named-entities (NEs), which is written by expertise. Similarly, the **H** in question answering data is constructed using both the question and the (in)correct answers.

Therefore, the research questions we can ask are:

1. Are these hypotheses really the ones people without expertise interested in?

2. Are hypotheses different if we construct them in other ways?

3. What would be representative negative hypotheses compared with the positive ones?

This particular setting of constructing RTE corpora may put a bias on the data being collected. Firstly, experts cannot represent all the people using the language; secondly, giving too much information restricts annotators' thinking; thirdly, the data are not balanced, since the negative cases include all the other textual semantic relations than entailment. Even though the RTE corpora have their own advantage, it is still interesting to see the other options.

On constructing the AMT corpus, we use Amazon's Mechanical Turk (MTurk)[16], online non-expert annotators (Snow et al., 2008). Instead of constructing the hypotheses targeted to IE or QA, we just ask the human annotators to come up with some facts they consider as relevant to the given text. For negative hypotheses, we change the instruction and ask them to write counter-factual but still relevant statements. In order to

---

[16]https://www.mturk.com/mturk/

narrow down the content of the generated hypotheses, we give a focused named-entity (NE) for each text to guide the annotators. The analysis of the results is performed by comparing the acquired data with the RTE challenge dataset.

7.3.1 Design of the Task

The basic idea of the task is to give the human annotators a paragraph of text with one highlighted named-entity and ask them to write some facts or counter-facts around it. In particular, we first preprocess an existing RTE corpus using a named-entity recognizer to mark all the named-entities appearing in both **T** and **H**. When we show the texts to Turkers, we highlight one named-entity and give them one of these two sets of instructions:

Facts: Please write several facts about the highlighted words according to the paragraph. You may add additional common knowledge (e.g., Paris is in France), but please mainly use the information contained in the text. But please **do not copy and paste**!

Counter-Facts: Please write several statements that are contradictory to the text. Make your statements about the highlighted words. Please use the information mainly in the text. Avoid using words like **not** or **never**.

Then there are three blank lines given for the annotators to fill in, either three facts or three counter-factual statements. For each task (called *human intelligence task* or HIT), we gather facts or counter-facts for five texts, and for each text, we ask three annotators to perform the task. We give Turkers one example as a guide along with the instructions.

7.3.2 Statistics of the Dataset

The texts we use in our experiments are the development set of the RTE-5 challenge (Bentivogli et al., 2009), and we preprocess the data using the Stanford named-entity recognizer (Finkel et al., 2005). In all, it contains 600 **T**-**H** pairs. We use the texts to generate facts and counter-facts, and hypotheses are used as references. In order to get reliable Turkers, we put our task online through CrowdFlower[17]. On average, we pay one

---

[17]http://crowdflower.com/

	Total	Average (per Text)
Extracted NEs		
Facts	244	1.19
Counter-Facts	121	1.11
Generated Hypotheses		
Facts	790	3.85
Counter-Facts	203	1.86

Table 7.14: The statistics of the (valid) data we collect

cent for each (counter-)fact to the Turkers and the data were collected within a few hours.

To get a sense of the quality of the data we collect, we mainly focus on analyzing the following three aspects: 1) the statistics of the datasets themselves; 2) the comparison between the data we collect and the original RTE dataset; and 3) the comparison between the facts and the counter-facts.

Table 7.14 show some basic statistics of the data we collect[18]. After excluding invalid and trivial ones[19], we acquire 790 facts and 203 counter-facts. In general, the counter-facts seem to be more difficult to obtain than the facts, since both the total number and the average number of the counter-facts are less than those of the facts. Notice that the NEs are not many since they have to appear in both **T** and **H**.

7.3.3   Analyses on the Dataset

The comparison between our data and the original RTE data is shown in Table 7.15[20]. The average length of the generated hypotheses is longer than the original hypotheses, for both the facts and the counter-facts.

---

[18]The *Total* column presents the number of extracted NEs and generated hypotheses and the *Average* column shows the average numbers per text respectively.

[19]Invalid data include empty string or single words; and the trivial ones are those sentences directly copied from the texts.

[20]The *Ave. Length* column represents the average number of words in each hypothesis; The *Ave. BoW* shows the average bag-of-words similarity compared with the text. The three columns on the right are all about the position of the NE appearing in the sentence, how likely it is at the head, middle, or tail of the sentence.

	Ave. Length	Ave. BoW	NE Position		
			Head	Middle	Tail
Original Entailment **H**s	7.6	0.76	46%	53%	1%
Facts	9.8	0.68	68%	29%	3%
Original Contradiction **H**s	7.5	0.72	44%	56%	0%
Counter-Facts	12.3	0.75	59%	38%	3%

Table 7.15: The comparison between the generated (counter-)facts and the original hypotheses from the RTE dataset

Counter-facts seem to be more verbose, since additional (contradictory) information is added.

Table 7.16 - Table 7.19 contain some (counter-)facts written by the Turkers. We ask them to write several for each highlighted NE, and only part of the results are shown here. For instance, example ID 425 (Table 7.18), Counter_Fact_1 can be viewed as the more informative but contradictory version of Fact_1 (and the original hypothesis). The average bag-of-words similarity scores are calculated by dividing the number of overlapping words of **T** and **H** by the total number of words in **H**. In the original RTE dataset, the entailed hypotheses have a higher BoW score than the contradictory ones; while in our data, facts have a lower score than the counter-facts. This may be caused by the greater variety of the facts than the counter-facts. Fact_1 of example ID 425 (Table 7.18) is almost the same as the original hypothesis, and Fact_2 of example ID 374 (Table 7.17) as well, though the latter has some slight differences which make the answer different from the original one. The NE position in the sentence is another aspect to look at. We find that people tend to put the NEs at the beginning of the sentences more than other positions, while in the RTE datasets, NEs appear in the middle more frequently.

In order to get a feeling of the quality of the data, we randomly sampled 50 generated facts and counter-facts and manually compared them with the original hypotheses. Table 7.20 shows that generated facts are easier for the systems to recognize, and the counter-facts have the same difficulty on average[21].

In order to reduce the subjectivity of evaluating the *difficulty* of the data by human reading, we follow the criteria that:

---

[21]The *Valid* column shows the percentage of the valid (counter-)facts; and other columns present the distribution of harder, easier cases than the original hypotheses or with the same difficulty.

ID: 16	Answer: **Contradiction**
Original Text	*The father of an Oxnard teenager accused of gunning down a gay classmate who was romantically attracted to him has been found dead, Ventura County authorities said today. **Bill McInerney**, 45, was found shortly before 8 a.m. in the living room of his Silver Strand home by a friend, said James Baroni, Ventura County's chief deputy medical examiner. The friend was supposed to drive him to a court hearing in his son's murder trial, Baroni said. McInerney's 15-year-old son, Brandon, is accused of murder and a hate crime in the Feb. 12, 2008, shooting death of classmate Lawrence "Larry" King, 15. The two boys had been sparring in the days before the killing, allegedly because Larry had expressed a romantic interest in Brandon.*
Original Hypothesis	***Bill McInerney** is accused of killing a gay teenager.*

NE_1: Bill McInerney	
Counter_Fact_1	*Bill McInerney is still alive.*

Table 7.16: Examples of facts compared with the original texts and hypotheses (ID: 16).

1. Abstraction is more difficult than extraction;

2. Inference involving several steps is more difficult than the direct entailment;

3. The more sentences in **T** are involved, the more difficult that **T-H** pair is.

Therefore, we view the Counter_Fact_1 in example ID 16 (Table 7.16) is more difficult than the original hypothesis, since it requires more inference than the direct fact validation. However, in example ID 374 (Table 7.17), Fact_1 is easier to be verified than the original hypothesis,

ID: 374	Answer: **Contradiction**
Original Text	*Other friends were not surprised at his death. "I wasn't surprised," said **George Stranahan**, a former owner of the **Woody Creek Tavern**, a favourite haunt of Thompson. "I never expected Hunter to die in a hospital bed with tubes coming out of him." Neighbours have said how his broken leg had prevented him from leaving his house as often as he had liked to. One neighbour and long-standing friend, Mike Cleverly, said Thompson was clearly hobbled by the broken leg. "Medically speaking, he's had a rotten year."*
Original Hypothesis	*The **Woody Creek Tavern** is owned by **George Stranahan**.*

NE_1: George Stranahan	
Fact_1	*George Stranahan spoke of Thompson's death.*
Fact_2	*George Stranahan once owned the Woody Creek Tavern.*
Counter_Fact_1	*George Stranahan was surprised by his friend's death.*
Counter_Fact_2	*Medically, George Stranahan's friend, Humter Thompson, had a great year.*
Counter_Fact_3	*George Stranahan fully expected Thompson to die in a hospital with tubes coming out of him.*

NE_2: Woody Creek Tavern	
Fact_1	*Woody Creek Tavern was previously owned by George Stranahan.*

Table 7.17: Examples of facts and counter-facts compared with the original texts and hypotheses (ID: 374).

ID: 425	Answer: **Entailment**
Original Text	*Merseyside Police concluded after a brief inspection that the controversial blog **Liverpool Evil Cabal** does not break criminal law. However the council officers continue to search for the editor. The blog has been blocked on computers controlled by Liverpool Direct Ltd, a company jointly owned by Liverpool City Council and British Telecom. The council's elected officials have denied ordering the block and are currently investigating its origin.*
Original Hypothesis	***Liverpool Evil Cabal** is the name of an online blog.*

NE_1: Liverpool Evil Cabal	
Fact_1	*Liverpool Evil Cabal is a web blog.*
Fact_2	*Liverpool Evil Cabal was a blog investigated by the Merseyside Police.*
Counter_Fact_1	*Liverpool Evil Cabal is a blog of Liverpool Direct Ltd.*
Counter_Fact_2	*Liverpool Evil Cabal is freed from the charges of law breaking.*

Table 7.18: Examples of facts and counter-facts compared with the original texts and hypotheses (ID: 425).

and same as those facts in example ID 506 (Table 7.19). Similar hypotheses (e.g., Fact_1 in example ID 425 and the original hypothesis) are treated as being at the same level of difficulty.

After the quantitive analysis, let's take a closer look at the examples. The facts are usually constructed by rephrasing some parts of the text (e.g., in ID 425, "after a brief inspection" is paraphrased by "investigated by" in Fact_2) or making a short summary (e.g., Fact_1 in ID 374, "George Stranahan spoke of Thompson's death."). For counter-facts, removing the negation words or changing into another adjective is one common choice, e.g., in ID 374, Counter_Fact_1 removed "n't" and Counter_Fact_3 changed "never" into "fully". Antonyms can also make

ID: 506	Answer: **Entailment**
Original Text	*At least 58 people are now dead as a result of the recent flooding in **Yemen**, and at least 20,000 in the country have no access to shelter. Five people are also reported missing. The Yemeni government has pledged to send tents to help the homeless. The flooding is caused by the recent heavy rain in Yemen, which came as a shock due to the fact that the country only receives several centimeters of rain per year.*
Original Hypothesis	*Heavy rain caused flooding in **Yemen**.*

NE_1: Yemen	
Fact_1	*58 people are dead in Yemen because of flooding.*
Fact_2	*5 people in Yemen are missing.*
Fact_3	*At least 58 people are dead in Yemen because of flooding.*

Table 7.19: Examples of facts compared with the original texts and hypotheses (ID: 506).

the contradiction, as "rotten" to "great" in Counter_Fact_2 in ID 374.

Example ID 506 in Table 7.19 is another interesting case. There are many facts about Yemen, but no valid counter-facts are generated. Furthermore, if we compare the generated facts with the original hypothesis, we find that people tend to give straightforward facts instead of entailments that involve several steps of reasoning[22].

To conclude this section, we show some preliminary results on testing a baseline RTE system on this dataset. For the sake of comparison, we extract a subset of the dataset, which is balanced on entailment and contradiction text pairs, and compare the results with the same system on the original RTE-5 dataset. The baseline system uses a simple BoW-based similarity measurement between **T** and **H** (Bentivogli et al., 2009)

	Valid	Harder	Easier	Same
Facts	76%	16%	24%	36%
Counter-Facts	84%	36%	36%	12%

Table 7.20: The comparison of the generated (counter-)facts with the original hypotheses

Datasets (Counter-/Facts)	RTE-5 (300/300)	Our Data (178/178)
All "YES"	50%	50%
BoW Baseline	57.5%	58.4%

Table 7.21: The results of baseline RTE systems on the data we collected, compared with the original RTE-5 dataset

and the results are shown in Table 7.21[23].

The results indicate that our data are slightly "easier" than the original RTE-5 dataset, which is consistent with our human evaluation on the sampled data (Table 7.20). However, it is still too early to draw conclusions based on the simple baseline tests.

7.4 Summary

In this chapter, we discussed several corpora. Firstly, we give an overview of several corpora annotated with different kinds of textual semantic relations, as well as the methodology of constructing them. Then we present our work on constructing two corpora, the TSR corpus and the AMT corpus.

For the TSR corpus, we extract text span pairs related by different discourse relations (from six broad relation groups) and annotate each pair with one of six semantic relations. Despite the fact that it is difficult to find totally clear-cut examples of semantic relations such as entailment or contradiction in naturally occurring examples of adjacent text spans, we do obtain a relatively high inter-annotator agreement. For the AMT corpus, we use MTurk to collect facts and counter-facts about the given NEs

---

[22]This may be caused by the design of our task.

[23]The *Counter-/Facts* row shows the number of the **T-H** pairs contained in the dataset; and the other scores in percentage are accuracy of the systems.

and texts. We discover that the generated hypotheses are not entirely the same as the original hypotheses in the RTE data, which provides us with alternative data from the non-expert annotators.

The data and annotation we obtain are different from the existing RTE corpora in the following aspects:

1. The TSR corpus contains naturally occurring data instead of constructively created ones;

2. The TSR corpus is more *balanced* than the RTE corpus or the MSR corpus in terms of semantic relations between two texts, which alleviates the problem of finding the representative negative examples;

3. The AMT corpus comes from the untrained annotators without much linguistic or NLP knowledge, and the data are assumed to be less artificial.

In sum, we are convinced that the two issues of the corpus construction mentioned at the beginning, the annotation scheme and the data collection method, are of great importance and needs to be further explored. It is difficult to draw reliable conclusions before the system evaluation. Therefore, all the five corpora described above are used in our experiments of the TSR recognition (Chapter 9). Before that, in the next chapter, we evaluate the impact of one of the proposed measurement, *relatedness*, on the RTE task first.

8 Textual Relatedness Recognition

In this chapter[1], we mainly focus on the textual relatedness recognition. As we have already discussed in Chapter 6 that, although *relatedness* is usually user-dependent, in practice, it may help with filtering out the noisy cases. It is also an intermediate step to achieve the classification of all textual semantic relations between two texts. We firstly introduce our meaning representation based on the semantic dependency graphs and then define a numerical measurement called *Textual Relatedness* between any pair of texts. It is linguistically-indicated and can be viewed as a weaker concept than the semantic similarity. In the experiments, we show that an alignment model based on the predicate-argument structures using this measurement can help an RTE system to recognize the UNKNOWN cases at the first stage, and contribute to the improvement of the overall performance as well. In addition, several heterogeneous lexical resources are tested, and different contributions from them are observed.

---

[1]Section 8.1 to Section 8.3 have been published in (Wang and Zhang, 2009), and it was a collaboration with Dr. Yi Zhang. Section 8.4 has been published in (Wang et al., 2009), and it was a collaboration with Dr. Yi Zhang and PD Dr. Günter Neumann.

8.1 Meaning Representation

Previously, RTE systems using semantic role labeling (SRL) have not shown very promising results, although SRL has been successfully used in many other NLP tasks, e.g., information extraction, question answering, and so on. According to our analysis of the data, there are mainly three reasons:

1. the limited coverage of the verb frames or predicates;

2. the undetermined relationships between two frames or predicates;

3. the dissatisfactory performance of an automatic SRL system

For instance, Burchardt et al. (2007) attempted to use FrameNet (Baker et al., 1998) for the RTE-3 challenge, but did not show substantial improvement. With the recent CoNLL shared tasks (Surdeanu et al., 2008, Hajič et al., 2009) focusing on semantic dependency parsing along with the traditional syntactic dependency parsing, more and more robust and accurate SRL systems are ready for use, especially for the predicate-argument structure (PAS) identification.

In order to obtain the PAS, we utilize an SRL system developed by Zhang et al. (2008). The SRL system is trained on the Wall Street Journal sections of the Penn Treebank using PropBank and NomBank annotation of verbal and nominal predicates, and relations to their arguments, and produces as outputs the semantic dependencies. The head words of the arguments (including modifiers) are annotated as a direct dependent of the corresponding predicate words, labeled with the type of the semantic relation (Arg0, Arg1 ..., and various ArgNs). Note that for the application of SRL in RTE task, the PropBank and NomBank notation appears to be more accessible and robust than the the FrameNet notation (with much more detailed roles or frame elements bond to specific verb frames).

As input, the SRL system requires syntactic dependency analysis. We use the open source MST Parser (McDonald et al., 2005), trained also on the Wall Street Journal Sections of the Penn Treebank, using a projective decoder with second-order features. Then the SRL system goes through a pipeline of 4-stage processing: predicate identification (PI) identifies words that evokes a semantic predicate; argument identification (AI) identifies the arguments of the predicates; argument classification

(AC) labels the argument with the semantic relations (roles); and predicate classification (PC) further differentiate different use of the predicate word. All components are built as maximal entropy based classifiers, with their parameters estimated by the open source TADM system[2], feature sets selected on the development set. Evaluation results from previous years' CoNLL shared tasks show that the system achieves state-of-the-art performance, especially for its out-domain applications.

Figure 8.1[3] and Figure 8.2 show the resulting semantic dependency graphs of the following example from the the RTE-4 test set (Giampiccolo et al., 2009),

T: *At least five people have been killed in a head-on train collision in north-eastern France, while others are still trapped in the wreckage.* ***All the victims are adults.***

H: ***A French train crash killed children.***

			A0		A1	
–Root–	All	the	victims	are	adults	.
	All	the	victim	be	adult	.
	PDT	DT	NNS	VBP	NNS	.
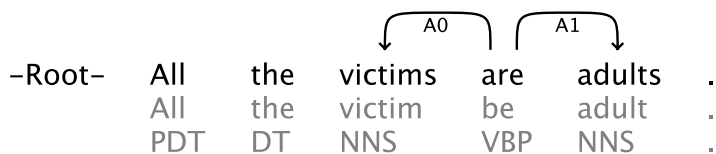
Figure 8.1: The semantic dependency graph of the second sentence of the Text

			A1	A0		A1	
–Root–	A	French	train	crash	killed	children	.
	a	French	train	crash	kill	child	.
	DT	JJ	NN	NN	VBD	NNS	.
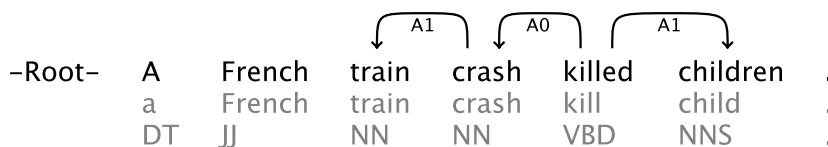
Figure 8.2: The semantic dependency graph of the Hypothesis

Before elaborating on the application of such a meaning representation to the RTE task, the next section firstly define the *relatedness* measurement. It is based on this meaning representation together with lexical semantic relations. Notice that although we only define the relatedness between **T** and **H**, in principle, the approach can be used to define other semantic relations. Actually, in Chapter 9, we use similar approaches to define several other measurements.

---

[2]`http://tadm.sourceforge.net/`

[3]Both Figure 8.1 and Figure 8.2 are generated by *whatswrong*: `http://code.google.com/p/whatswrong/`. And so as Figure 9.1 and Figure 9.2 in Chapter 9.

8.2   Relatedness Definition

As we mentioned in Chapter 6, we break down the three-way classification into a two-stage binary classification. Furthermore, we treat the first stage as a subtask of the main task, which determines whether **H** is related to **T**. Similar to the probabilistic entailment score, we use a relatedness score to measure such a relationship. According to the definition of textual entailment, **H** should be fully entailed by **T**. We also make this relatedness relationship asymmetric. Roughly speaking, this *relatedness* function $R(\mathbf{T}, \mathbf{H})$ can be described as whether or how relevant **H** is to some part of **T**. The relevance can be realized as string similarity, semantic similarity, or being co-occurred in similar contexts.

Although the term, *Textual Relatedness*, has not been widely used by the community (as far as we know), many researchers have already incorporated modules to tackle it, which are usually implemented as an alignment module before the inference/learning module is applied. For example, Padó et al. (2009b) mentioned two alignment modules, one is a phrase-based alignment system called MANLI (MacCartney et al., 2008), and the other is a stochastic aligner based on dependency graphs. Siblini and Kosseim (2009) performed the alignment on top of two ontologies. We follow this line of research but on another level of representation, i.e., the predicate-argument structures (PAS), together with different lexical semantic resources.

After semantic parsing described in the previous section, we obtain a PAS for each sentence. On top of it, we define a *predicate-argument graph* (PAG), the nodes of which are predicates, arguments or sometimes both, and the edges of which are labeled semantic relations. Notice that each predicate can dominate zero, one, or more arguments, and each argument has one or more predicates which dominate it. Furthermore, the graph is not necessarily fully connected. Thus, the $R(\mathbf{T}, \mathbf{H})$ function can be defined on the dependency representation as follows: if the PAG of **H** is semantically relevant to part of the PAG of **T**, **H** is semantically relevant to **T**.

In order to compare the two graphs, we further reduce the alignment complexity by breaking the graphs into sets of trees. Two types of decomposed trees are considered: one is to take each predicate as the root of a tree and arguments as children nodes, and the other is to take each argument as root and their governing predicates as children nodes. We name them as *Predicate Trees* (P-Trees) and *Argument Trees* (A-Trees)

respectively. To obtain the P-Trees, we enumerate each predicate, find all the arguments which it directly dominates, and then construct a P-Tree. The algorithm to obtain A-Trees works in the similar way. Finally, we have a set of P-Trees and a set of A-Trees for each PAG, both of which are simple trees with depth of one.
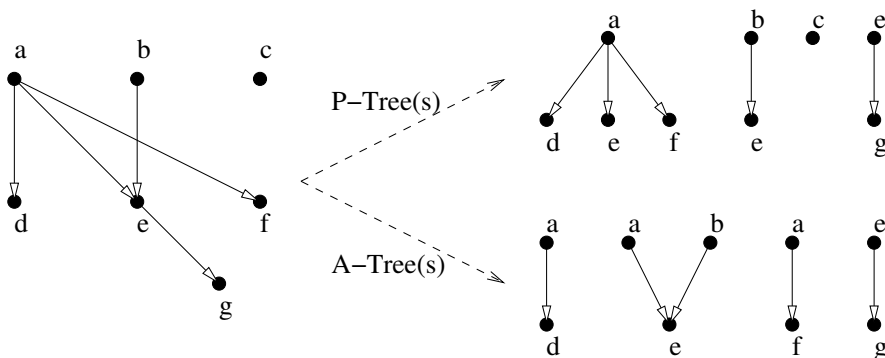

Figure 8.3: Decomposition of predicate-argument graphs (left) into P-Trees (right top) and A-Trees (right bottom)

Figure 8.3 shows examples of how the P-Trees and A-Trees algorithms work. Notice that we do not consider cross-sentential inference, instead, we simply take the union of tree sets from all the sentences.

Figure 8.4 illustrates the PAG for both **T** and **H** after semantic parsing, and the resulting P-Trees and A-Trees after applying the decomposition algorithm.

Formally, we define the relatedness function for a **T**-**H** pair as the maximum value of the relatedness scores of all pairs of trees in **T** and **H** (P-trees and A-trees).

$$R(T, H) = \max_{1 \leq i \leq r, 1 \leq j \leq s} \left\{ R(\mathit{Tree}_{T_i}, \mathit{Tree}_{H_j}) \right\}$$

In order to compare two P-Trees or A-Trees, we further define each predicate-argument pair contained in a tree as a semantic dependency triple. Each semantic dependency triple contains a predicate, an argument, and the semantic dependency label in between, in the form of

$$\langle \mathit{Predicate}, \mathit{Dependency}, \mathit{Argument} \rangle$$

Then we define the relatedness function between two trees as the minimum value of the relatedness scores of all the triple pairs from the two trees:
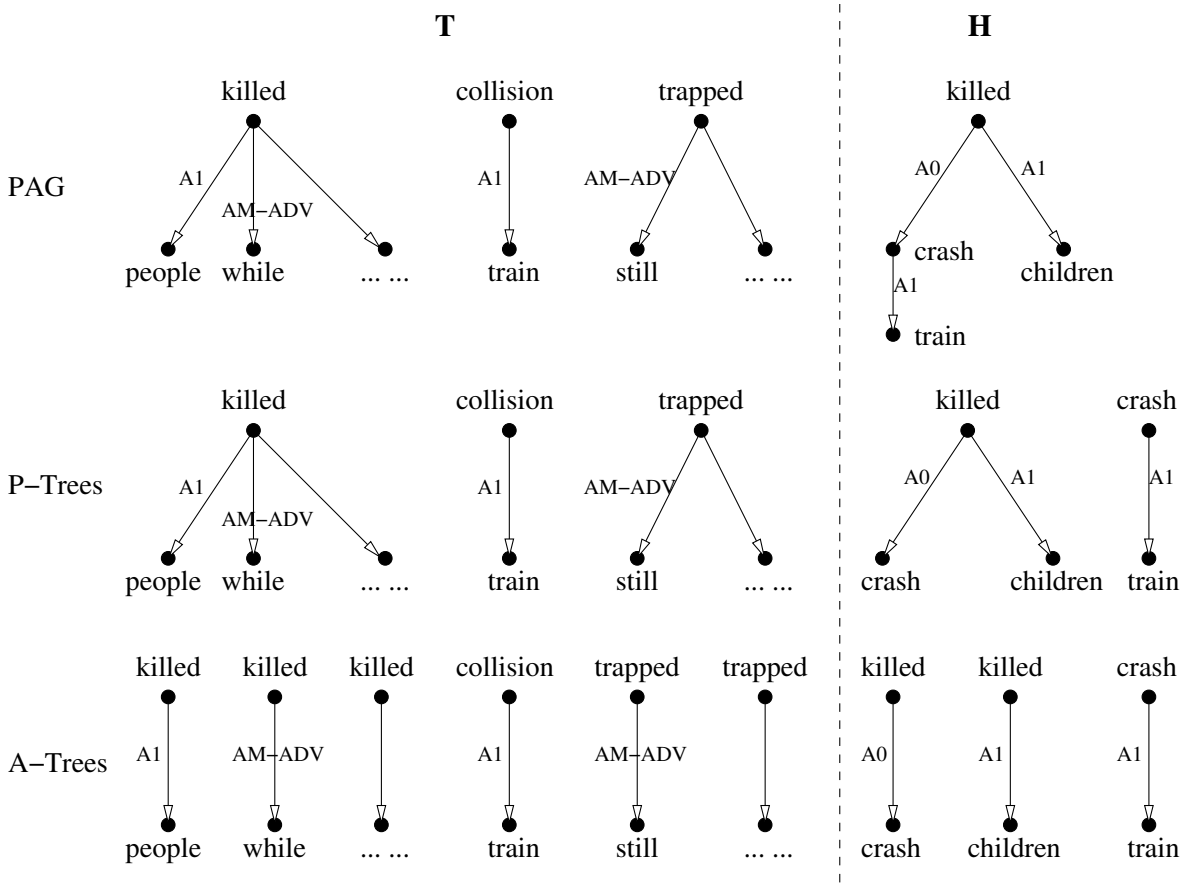
Figure 8.4: Predicate-argument graphs and corresponding P-Trees and A-trees of the **T-H** pair.

$$R(\mathit{Tree_T}, \mathit{Tree_H}) = \min_{1 \le i \le n, 1 \le j \le m} R(\langle P_T, D_{T_i}, A_{T_i} \rangle, \langle P_H, D_{H_j}, A_{H_j} \rangle)$$

For the relatedness function between two semantic dependency triples, we define the following two settings: the FULL match and the NOTFULL match. Both match types require that the predicates are related. The former means both the dependencies and the arguments are related, while the latter only requires the dependencies to be related.

$$R(\langle P_T, D_T, A_T \rangle, \langle P_H, D_H, A_H \rangle) = \\ \begin{cases} \text{Full} & R(P_T, P_H) = R(D_T, D_H) = R(A_T, A_H) = 1 \\ \text{NotFull} & R(P_T, P_H) = R(D_T, D_H) = 1 \\ \text{Other} & \text{Otherwise} \end{cases}$$

Now, the only missing components in our definition is the related-

ness functions between predicates, arguments, and semantic dependencies. Fortunately, we could use the results from the research on semantic relatedness in lexical semantics. Therefore, these functions can be realized by different string matching algorithms and/or lexical resources. Since 'relevance' can be defined in multiple ways, apart from the string matching of the lemmas, we also incorporate various resources, from distributionally collected ones to hand-crafted ontologies. We choose VerbOcean (Chklovski and Pantel, 2004) to obtain the relatedness between predicates (after using WordNet (Fellbaum, 1998) to change all the nominal predicates into verbs) and use WordNet for the argument alignment. For the verb relations in VerbOcean, we consider all of them as related; and for WordNet, we not only use the synonyms, hyponyms, and hypernyms, but antonyms as well. Consequently, we simplify these basic relatedness functions into a binary decision. If the corresponding strings are matched or the relations mentioned above exist, the two predicates, arguments, or dependencies are related; otherwise, they are not.

In addition, the Normalized Google Distance (NGD) (Cilibrasi and Vitanyi, 2007) is applied to both cases[4]. As for the comparison between dependencies, we simply apply string matching, except for modifier labels, which we treat the same[5]. All in all, the main idea here is to incorporate both distributional semantics and ontological semantics in order to see whether their contributions are overlapping or complementary. In practice, we use the empirical value 0.5 as the threshold. Below the threshold means they are related, otherwise not. In order to achieve better coverage, we use the OR operator to connect all the relatedness functions above, which means, if any of them holds, the two items are related.

8.3 Experiments

In order to evaluate our method, we have set several experiments. The baseline system here is a simple Naive Bayes classifier with a feature set containing the Bag-of-Words (BoW) overlapping ratio between **T** and **H**, and also the syntactic dependency overlapping ratio. The feature model

---

[4]You may find the NGD values of all the content word pairs in RTE-3, RTE-4, and RTE-5 datasets at `http://www.coli.uni-sb.de/~rwang/resources/RTE3_RTE4_NGD.zip` and `http://www.coli.uni-sb.de/~rwang/resources/RTE5_NGD.zip`.

[5]This is mainly because it is more difficult for the SRL system to differentiate modifier labels than the complements.

combines two baseline systems proposed by previous work, which gives quite competitive performance. Since the main goal of this work is to show the impact of the PAS-based alignment module, we do not compare our results with other RTE systems (in fact, the baseline system already outperforms the average accuracy score of the RTE-4 challenge).

The main data set used for testing here is the RTE-4 data set with three-way annotations (500 entailment **T**-**H** pairs (E), 150 contradiction pairs (C), and 350 unknown pairs (U)). The results on RTE-3 data set (combination of the development set and test set, in all, 822 E pairs, 161 C pairs, and 617 U pairs) is also shown, although the original annotation is two-way and the three-way annotation was done by different researchers after the challenge[6].

We firstly show the performance of the baseline systems, followed by the results of our PAS-based alignment module and its impact on the whole task. After that, we also give more detailed analyses of our alignment module, according to different lexical relatedness measurements.

8.3.1   Baselines

The baseline systems used here are based on the overlapping ratio of words and syntactic dependencies between **T** and **H**. For the word overlapping ratio, we calculate the number of overlapping tokens between **T** and **H** and normalize it by dividing it by the number of tokens in **H**. The syntactic dependency overlapping ratio works similarly: we calculate the number of overlapping syntactic dependencies and divide it by the number of syntactic dependencies in **H**, i.e., the same as the number of tokens. Enlightened by the relatedness function, we also allow either FULL match (meaning both the dependencies and the parent tokens are matched), and NOTFULL match (meaning only the dependencies are matched). Here we only use string match between lemmas and syntactic dependencies. Table 8.1 presents the performance of the baseline system.

The results show that, even with the same classifier and the same feature model, with a proper two-stage strategy, it can already achieve better results than the three-way classification. Note that the first strategy that corresponds to the traditional two-way annotation of the RTE

---

[6]The annotation of the development set was done by students at Stanford, and the annotation of the test set was done as double annotation by NIST assessors, followed by adjudication of disagreements. Answers were kept consistent with the two-way decisions in the main task gold answer file.

Strategies	Three-Way	Two-Stage		
	$E/C/U$	E/CU $\rightarrow E/C/U$	C/EU $\rightarrow C/E/U$	U/EC $\rightarrow U/E/C$
Accuray	53.20%	50.00%	53.50%	54.20%
Upper Bound	/	82.80%	68.70%	84.90%

Table 8.1: Performances of the baselines

task is not so successful. Our explanation here is that the BoW method (even with syntactic dependency features) is based on overlapping information shared by **T** and **H**, which essentially means the more information they share, the more relevant they are, instead of being more similar or the same. Therefore, for the "$ECU \rightarrow E/CU$" setting, methods based on overlapping information are not the best choice, while for "$ECU \rightarrow U/EC$", they are more appropriate. To our best knowledge, the detailed comparison between these strategies has not been fully explored, let alone the impact of the linguistic motivation behind the strategy selection.

In addition, the upper bound numbers show the accuracy when the first-stage classification is perfect, which give us an indication of how far we can go. The lower upper bound for the second strategy is mainly due to the low proportion of the C cases (15%) in the data set, while for the other two both show large space for improvement.

8.3.2   The PAS-based Alignment Module

In this subsection, we present a separate evaluation of our PAS-based alignment module. As we mentioned before (Section 8.2), there are several parameters to be tuned in our alignment algorithm: a) whether the relatedness function between P-Trees asks for the FULL match; b) whether the function for A-Trees asks for the FULL match; and c) whether both P-Trees and A-Trees being related are required or either of them holds is enough. Since they are all binary values, we use a three-character code to represent each setting, e.g., [FFO][7] means *either* P-Trees are FULL matched *or* A-Trees are FULL matched. The performances of different settings of the module are shown in the following Precision-Recall figure 8.5,

---

[7]F stands for FULL, and O stands for OR. Other letters are, N stands for NOTFULL, and A stands for AND.
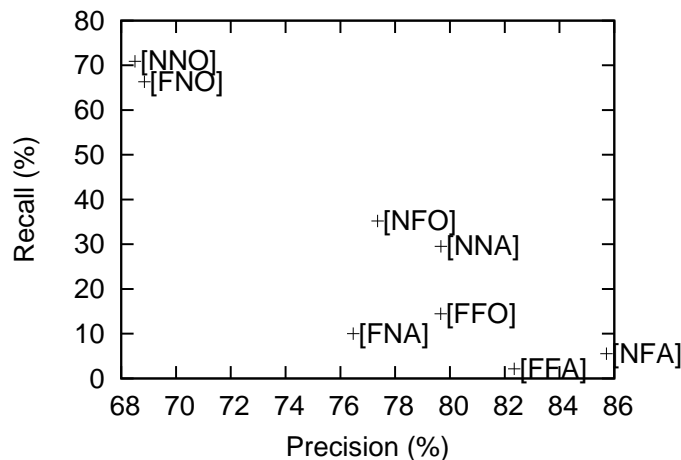
Figure 8.5: Precision and recall of different alignment settings

Since we combine this module with the baseline system and it is integrated as the first-stage classification, the F1 scores are not indicative for selecting the best setting. Intuitively, we may prefer higher precision than recall.

There is one limitation of our method. If some important predicates or arguments in **H** are not (correctly) identified by the SRL system, fewer P-Trees and A-Trees are required to be related to some part of **T**, thus, the relatedness of the whole pair is high, leading to false positive cases.

8.3.3   Impact on the Final Results

The best settings for RTE-3 data set is [NNA] and for RTE-4 data set is [NFO], which are both in the middle of the setting range shown in the previous figure 8.5.

As for the integration of the PAS-based alignment model with our BoW-based baseline, we only consider the third two-stage classification strategy in Table 8.1. Other strategies are also interesting to try, however, the proposed alignment algorithm exploits relatedness between **T** and **H**, which may not be fine-grained enough to detect entailment or contradiction. A new alignment strategy needs to be designed to explore other strategies. Thus, we believe that the alignment algorithm based on PAS (and other methods based on overlapping information between **T** and **H**) is suitable for the $U/EC \rightarrow U/E/C$ classification strategy.

Table 8.2 shows the final results and Table 8.3 shows the results at

Datasets [Systems]	Three-Way (Baseline1)	Two-Stage	
		(Baseline2)	(SRL+Baseline2)
RTE-3 [NNA]	52.19%	52.50%	53.69%(2.87%↑)
RTE-4 [NFO]	53.20%	54.20%	56.60%(6.39%↑)

Table 8.2: Results on the whole datasets

Datasets [Systems]	Baseline2	SRL+Baseline2	SRL
RTE-3 [NNA]	59.50%	60.56%(1.78%↑)	70.33%
RTE-4 [NFO]	67.10%	70.20%(4.62%↑)	79.67%

Table 8.3: System performances at the first stage

the first stage classification. The first observation is that the improvement of accuracy on the first stage of the classification can be preserved to the final results. And our PAS-based alignment module can help, though there is still large space for improvement. Compared with the significantly improved results on RTE-4, the improvement on RTE-3 is less obvious, mainly due to the relatively lower precision (70.33% vs. 79.67%) of the alignment module itself.

Also, we have to say that the improvement is not as big as we expected. There are several reasons for this. Besides the limitation of our approach mentioned in the previous section, the predicates and arguments themselves are too sparse to convey all the information we need for the entailment detection. In addition, the baseline is quite strong for this comparison, since the PAS-based alignment module relies on the overlapping words in the first place. There are quite a few pairs solved by both the main approach and the baseline. Then, it is interesting to take a closer look at the lexical resources used in the main system, which is another advantage over the baseline.

8.3.4 Impact of the Lexical Resources

We did an ablation test of the lexical resources used in our alignment module. Recall that we have applied three lexical resources, VerbOcean for the predicate relatedness function, WordNet for the argument relatedness function, and Normalized Google Distance for both. Table 8.4 shows the performances of the system without each of the resources,

The results clearly show that each lexical resource does contribute

Data Sets [Systems]	SRL +Baseline2	Without VerbOcean	Without NGD	Without WordNet
RTE-3 [NNA]	53.69%	53.19% (0.93%↓)	53.50% (0.35%↓)	52.88% (1.51%↓)
RTE-4 [NFO]	56.60%	56.00% (1.06%↓)	56.10% (0.88%↓)	55.70% (1.59%↓)

Table 8.4: Impact of the lexical resources

some improvement to the final performance of the system and it confirms the idea that combining lexical resources that are acquired in different ways can be valuable. For instance, at the beginning, we expected that the relationship between "people" and "children" could be captured by WordNet, but in fact, it cannot. Fortunately, the NGD has a quite low value of this pair of words (0.21), which suggests that they occur together quite often, or in other words, they are *related*.

One interesting future direction is to substitute the OR connector between these lexical resources with an AND operator. Thus, instead of using them to achieve higher coverage, it is also interesting to know whether they can be filters for each other by increasing the precision.

8.4   Extension of the Approach

One may notice that the main system (based on PAS) only uses the semantic dependency graph, while the backup system is only based on the syntactic dependency tree. Why not combine them into a joint representation? It is observed that the PAS can effectively deal with (some) syntactic variations like active/passive voice transformation, nominalization of the events, and so on. However, the semantic dependency fails to reach the syntactic object inside each prepositional phrase, which is of great importance for matching key information between **T** and **H**. Moreover, sentence-based syntactic and semantic dependency analysis suffer from unsolved cross-sentential co-references, when **T** contains more than one sentence, and the problem becomes even more severe if the length of **T** increases (as in RTE-5, cf. Section 7.1.1).

In order to solve these two problems:

- We combine syntactic and semantic dependency structure into a connected graph, achieving a new joint representation which can better

capture the overlapping information between **T** and **H**.

- We also use a co-reference resolver to group different mentionings of the same entity together to share the information between sentences.

In the following, we firstly introduce this joint representation and then present the evaluation results in the context of our participation in the RTE-5 challenge, which we ranked 2nd out of the 20 participants.

8.4.1 Joint Representation

Before introducing the joint representation, let us first take a closer look at the problems of the pure syntactic or semantic dependency structure as the meaning representation.
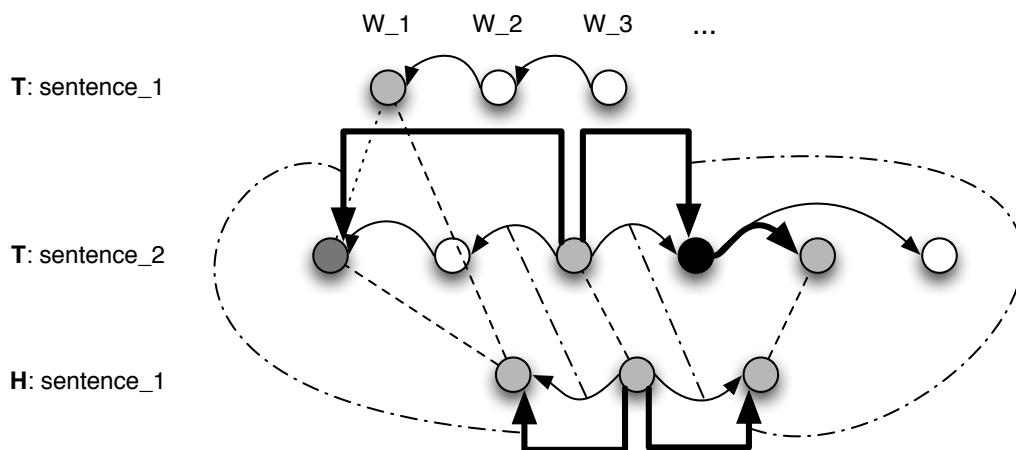
Figure 8.6: Example of an alignment based on the joint representation

Figure 8.6 shows an abstract example of an alignment between **T** and **H**. Each circle represents a word/token in the sentence; circles with the same grayscale are aligned word pairs; the dark gray circle represents a co-reference of the light gray circle in sentence 1 of **T**; each arrow represents a dependency between two words, either a syntactic dependency (curved) or a semantic dependency (orthogonal); a dashed line means an alignment between two words, and a dash-dotted line means an alignment between two dependencies.

We simplified **H** into a concise sentence with only three words, e.g., a predicate with a subject and an object, while **T** contains two sentences (hence $\mathbf{T}_1$ and $\mathbf{T}_2$) and more information (ir)relevant to **H**. We assume

that $\mathbf{T}_2$ is aligned with $\mathbf{H}$ with more overlapping words (denoted by the circles with the light gray). Besides the word alignment, we also check the overlapping syntactic dependency triples (i.e., <word, relation, word>), but we observe that these overlapping syntactic dependency triples cannot help us to reach the aligned words (on the syntactic dependency tree). Therefore, we need to go one level deeper to the semantic dependencies.

Although the left-hand side is fully aligned by semantic dependencies, the light gray circle in $\mathbf{T}_2$ on the right-hand side still cannot be reached. The black circle here is the end of the semantic dependency graph, and usually it is realized as a preposition. Consequently, we take the syntactic dependency into account, which links the black circle to the light gray circle and it can be used as a *backup* link for the semantic dependencies. Therefore, the joint representation consists of two parts: 1) the semantic dependencies (which can be a bag of isolated graphs in some cases); and 2) the syntactic dependencies connecting the content words, where the semantic graph ends at functional words. This is marked in bold in Figure 8.6.

The dark gray circle denotes a co-reference of the light gray circle in $\mathbf{T}_1$. This occurs more frequently, if the text of $\mathbf{T}$ becomes longer. Since we can easily find the alignment between the first word of $\mathbf{H}$ and the first word in $\mathbf{T}_1$, the alignment can be potentially passed to the first word in $\mathbf{T}_2$ (which is, for example, a pronoun). Therefore, we apply a co-reference resolution toolkit, BART (Versley et al., 2008) to gather such cross-sentential references. In short, this resolver assigns a label for a bag of different mentionings of the same entity it discovers in the text and we just group all the mentionings together (according to the labels) for the word alignment module.

Based on these two processes, we can thus integrate all the information (under a certain discourse) into one unified representation, both horizontally (from different sentences) and vertically (from different levels of linguistic analyses).

The advantage of such a representation has been shown in the previous example (repeated here),

T: *At least five people have been killed in a head-on train collision **in north-eastern France**, while others are still trapped in the wreckage. All the victims are <u>adults</u>.*

H: *A **French** train crash killed <u>children</u>.*

This is an example of a contradiction, where the only contradictory

part lies on "adults" in **T** and "children" in **H** (shown with underline). Although this pair can already be solved by the previous approach based purely on the semantic dependency graphs, notice that "in north-eastern France" in **T** and "French" in **H** (shown in bold) cannot be aligned by only syntactic or semantic dependencies, because the AM-LOC argument of the predicate "collision" is the preposition "in", and "France" cannot be reached on the semantic dependency graph. The link from the preposition "in" to the object "France" is a syntactic dependency, which is included by the joint representation. Similarly, the active/passive voice transformation can also be captured. For example,

T: ***Yigal Amir***, *the student who **assassinated** Israeli Prime Minister Yitzhak Rabin, ...*

H: *Yitzhak Rabin was killed **by Yigal Amir**.*

"Yigal Amir" in **T** is linked to "assassinated" via semantic dependency (the syntactic dependency is not direct); while "Yigal Amir" in **H** is under the preposition "by" on the syntactic tree, but hidden on the semantic dependency graph.

8.4.2 Experiments

We participated in the RTE-5 challenge with this system. Therefore, the following experiments were conducted in the context of the challenge, both the dataset and the results. The RTE-5 dataset contains 2400 **T-H** pairs, half in the development set and half in the test set. The annotation is three-way: ENTAILMENT 50%, CONTRADICTION 15%, and UNKNOWN 35%.

We have submitted three runs for the challenge, and both the syntactic parser and the semantic role labeler are the same as the previous experiments:

- Run1: The original system;

- Run2: The extended system (FFO[8]);

- Run3: The extended system (NNA).

The three-way results and the ablation test results are shown in the following Table 8.5:

---

[8]It means either predicate trees ask for a full match or argument trees ask for a full match. More detailed can be found in Section 8.3.2.

Runs	Main	Without VerbOcean	Without WordNet	Without Both
Run1	50.7%	50.5%	50.7%	50.5%
Run2	63.7%	63.2%	63.3%	63.0%
Run3	63.5%	63.3%	63.3%	63.3%

Table 8.5: Official results of the three-way evaluation

Compared with the original system (Run1), the improvement of extended systems Run2 and Run3 is obvious. We attribute it to the two improvements we made: 1) the joint representation has its advantage over the pure syntactic or semantic dependency structure; and 2) the co-reference resolution in the longer texts is effective.

We also calculate the confusion matrix for the three-way submission Run2, the best three-way setting (Table 8.6).

Run2		Gold-Standard			
		Entailment	Contradiction	Unknown	Total
System	Entailment	238	60	77	375
	Contradiction	4	**21**	10	35
	Unknown	58	9	123	190
	Total	300	**90**	210	600

Table 8.6: Confusion matrix of the Run2 submission

Although the system confuses between many Entailment and Unknown cases, the most serious problem seems to be the contradiction recognition, whose recall is the lowest (21/90=23.3%). In fact, this difficulty has been mentioned in previous research (de Marneffe et al., 2008).

Finally, we present our two-way results, both on the traditional two-way classes (Table 8.7) and related vs. Unknown classes (Table 8.8).

Runs	Main	Without VerbOcean	Without WordNet	Without Both
Run1	62.5%	62.5%	62.7%	62.5%
Run2	66.8%	66.5%	66.7%	66.3%
Run3	68.5%	68.3%	68.3%	68.3%

Table 8.7: Results of the two-way evaluation: Entailment vs. others

These two tables clearly show that all our runs do well for relatedness

Runs	Main	Without VerbOcean	Without WordNet	Without Both
Run1	74.0%	73.7%	73.8%	73.7%
Run2	74.3%	73.7%	73.8%	73.5%
Run3	72.3%	72.2%	72.2%	72.2%

Table 8.8: Results of the two-way evaluation: UNKNOWN vs. others

recognition, which meets our original goal. The overall improvement from the worst to the best results on the traditional two-way annotation is less than the three-way annotation (6% vs. 13%).

8.5   Summary

In this chapter, we test our idea of casting the three-way RTE problem into a two-stage binary classification task. We apply an SRL system to derive the predicate-argument structure of the input sentences, and propose ways of calculating semantic relatedness between the shallow semantic structures of **T** and **H**. The experiments show improvements in the first-stage classification, which accordingly contribute to the final results of the RTE task.

In addition, we also extend the meaning representation of the system in two ways: 1) by using a joint representation of syntactic and semantic dependencies; and 2) by resolving co-references across sentences. In order to further improve relatedness recognition, we can apply almost all the resources used before for entailment recognition here, e.g., the DIRT rules (Lin and Pantel, 2001) (Chapter 5) or other paraphrase resources (Callison-Burch, 2008).

More importantly, relatedness recognition is an intermediate step toward classifying all the semantic relations between texts. Apart from the relatedness, we also want to see whether the PAS can help differentiating ENTAILMENT and CONTRADICTION. For instance, the semantic dependency of negation (AM-NEG) may be helpful for the contraction recognition. We should also consider the directionality of entailment, which is a different case of a bi-directional paraphrase. Both issues will be discussed in the next chapter.

Furthermore, instead of the rule-based approach, we can also treat all alignments as features for a machine-learning-based classifier. These all

together lead to our multi-dimensional approach for the textual semantic relation recognition.

9 Textual Semantic Relation Recognition

This chapter[1] is about the recognition of textual semantic relations (TSRs) between two texts. We start with a revisit of the meaning representation described in the previous chapters and make a generalization. Then we introduce a multi-dimensional classification approach, including *relatedness* as one of the dimensions. The other two are *inconsistency* and *inequality*. We evaluate our approach on the datasets we presented in Chapter 7, and show that this is a generalized approach for RTE, paraphrase identification, and other TSR recognition tasks.

---

[1]Section 9.1 to Section 9.3 have been published in (Wang and Zhang, 2011), and it was a collaboration with Dr. Yi Zhang.

9.1   Meaning Representation Revisited

In the previous chapter, we have shown a meaning representation based on the predicate-argument structure (PAS) as well as the joint representation combining it with the syntactic dependency tree. The joint representation has advantages over the single ones both rationally and empirically. However, the combination strategy was a bit ad hoc.

Figure 9.1 shows the resulting syntactic dependency trees of the following entailment example and the semantic dependency graphs are shown in Figure 9.2.

T: *Devotees of the market question the value of the work national service would perform.*
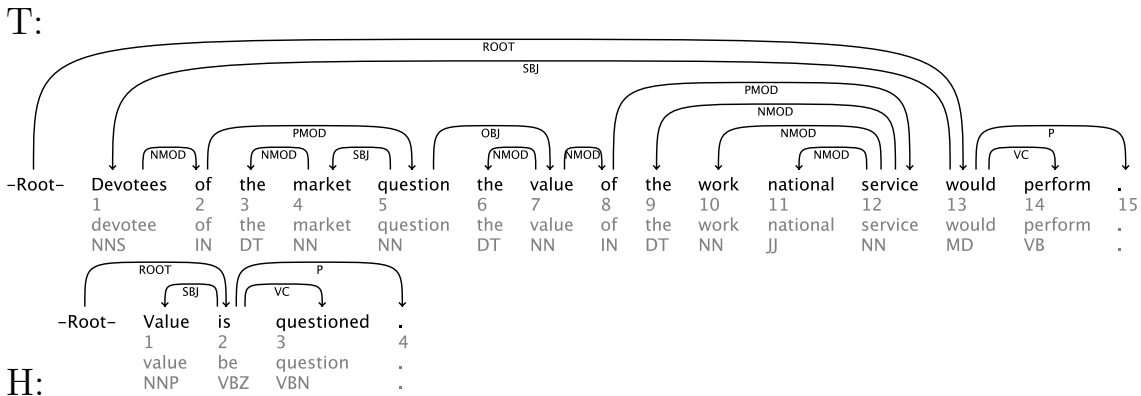
H: *Value is questioned.*

Figure 9.1: Syntactic dependency of the example **T-H** pair by Malt-Parser.
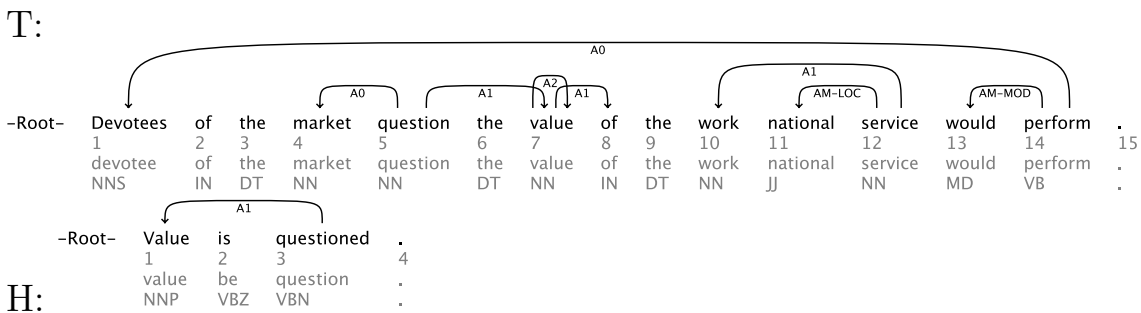

Figure 9.2: Semantic dependency of the example **T-H** pair by MaltParser and our SRL system.

If we assume that the meaning of the two texts is approximated by these four trees/graphs, the task is casted into verifying whether those (syntactic/semantic) dependency relations in **H** also appear in **T**. The procedure can be summarized into three steps:

1. Extracting the dependency triples in **H**, $\{DEP_H\}$;

2. Using the word pairs as anchors to find the corresponding dependency triples/paths in **T**, $\{DEP_T\}$;

3. Comparing the two sets, $\{DEP_T\}$ and $\{DEP_H\}$.

Basically, the meaning representation used in all the approaches we presented so far can be formalized into the following framework:

- The TACTE system (Chapter 4): $\{DEP_H\}$ is the set of all the event time pairs in **H** and $\{DEP_T\}$ contains those in **T**.

- The extended TACTE system (Section 4.7): $\{DEP_H\}$ is the set of all the event tuples in **H** and $\{DEP_T\}$ contains those in **T**.

- The RTE system with DIRT (Chapter 5): $\{DEP_H\}$ and $\{DEP_T\}$ are represented in the tree skeletons (Section 5.4.2).

- The *Relatedness* system (Chapter 8): $\{DEP_H\}$ contains all predicate-argument triples in **H** and $\{DEP_T\}$ has those in **T**.

- The extended *Relatedness* system (Section 8.4): $\{DEP_H\}$ is the same as the previous one and $\{DEP_T\}$ contains the dependency paths on the joint representation.

Notice that all the dependency triples come from the four graphs we showed before (Figure 9.1 and Figure 9.2), the generalization of the approach is just the generalization of the selection procedure. Furthermore, since it is not necessary that we can always find a direct dependency relation in **T** between the same word pair, we need to traverse the dependency tree or graph to find the dependency paths instead.

In general, we treat all the dependency trees and graphs as undirected graphs with loops, but keep records of the edge directions we traverse. We consider the following three cases:

Syntactic Dependency Tree We simply traverse the tree and find the corresponding dependency path connecting the two words;

Semantic Dependency Graph We use Dijkstra's algorithm (Dijkstra, 1959) to find the shortest path between the two words;

Joint Dependency Graph We assign different weights to syntactic and
semantic dependencies and apply Dijkstra's algorithm to find the
shortest path (with the lowest cost).

For the example shown above, we firstly find the dependency triples
in H, excluding those containing stop words. Then, the extracted syntactic dependency triples of the example **T**-**H** pair are ϕ, since the only
content words "value" and "questioned" have no direct syntactic dependency (Figure 9.1). The extracted semantic dependency triples are
<"questioned", "A1", "value"> (Figure 9.2). After that, we use the
word pairs contained in the extracted dependency triples as anchors to
find the corresponding dependency paths in **T**, in this case, <"question",
"A1", "value">.

In fact, this approach can be easily adapted to other meaning representations, if it can be transformed into the graph representation. In
the work presented in this chapter, we stay with the *Joint Dependency
Graph*.

9.2   System Description

As shown in Figure 9.3, the system consists of three procedures:

1. Preprocessing: We parse the input **T**-**H** pair and obtain both the
   syntactic dependency tree and the semantic dependency graph (details contained in the experiment part, Section 9.3.2).

2. Meaning Representation: We collect all the dependency triples (i.e.,
   <word, dependency relation, word>) in **H**, after excluding the stop
   words, and for each pair of words, we extract the corresponding dependency path in **T** (Section 9.1).

3. Feature-Based Classification: We extract features from the meaning representation and use an SVM-based classifier to determine the
   semantic relation (the rest of this section).

In the rest of this section, we elaborate on the feature extraction and
the TSR recognition.
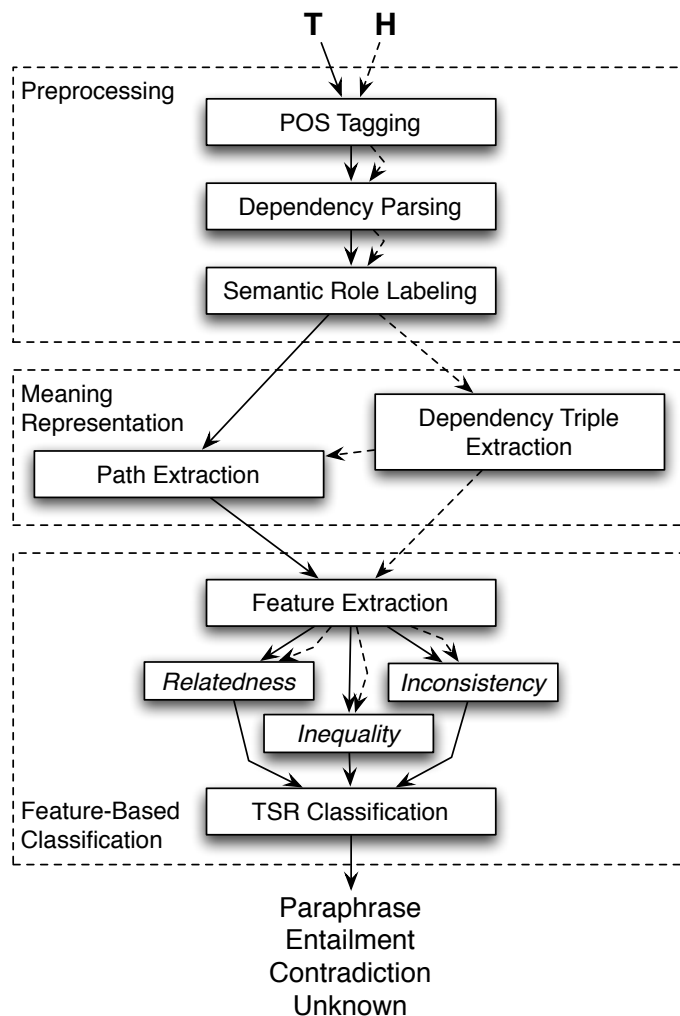
Figure 9.3: Workflow of the system

9.2.1 Feature Extraction

For the features, we firstly check whether there are dependency triples extracted from **H** as well as whether the same words can be found in **T**. Only if the corresponding dependency paths are successfully located in **T**, we can extract the following features.

The direction of each dependency relation or path is useful. We use a boolean value to represent whether the **T**-path contains dependency relations with different directions of the **H**-path. For instance, in Figure 9.2, if we extract the path from "market" to "value", the directions of the dependency relations contained in the path are ← and →, one of which is inconsistent with the dependency relation in **H**.

We define the length of one dependency path as the number of depen-

dency relations contained in the path. Thus, the dependency triple can be viewed as a dependency path which has length one, in other words, **H** only has dependency paths of length one, but the lengths of the dependency paths in **T** vary. If the length of the **T**-path is also one, we can directly compare the two dependency relations; otherwise, we compare each of the dependency relations contained in the **T**-path with the **H**-path one by one. Enlightened by Wang and Neumann (2007a), we exclude some dependency relations like "CONJ", "COORD", "APPO", etc., heuristically, since usually they do not change the relationship between the two words at both ends of the path.

By comparing the **T**-path with **H**-path, we mainly focus on two values, the category of the dependency relation (e.g., syntactic dependency vs. semantic dependency) and the content of the dependency relation (e.g., "A1" vs. "AM-LOC"). We also incorporate the string value of the dependency relation pair and make it boolean depending on whether it occurs or not.

	H_Null?	T_Null?	Dir	Multi?	Dep_Same?	Rel_Sim?	Rel_Same?	Rel_Pair
Syn Dep		+	+	+			+	+
Sem Dep	+	+	+	+		+	+	+
Joint	+	+	+	+	+	+	+	+

Table 9.1: Feature types of different settings of the system

Table 9.1 summarizes the feature types we extract from each **T-H** pair. *H_Null?* means whether **H** has dependencies; *T_Null?* means whether **T** has the corresponding paths (using the same word pairs found in **H**); *Dir* is whether the direction of the path **T** the same as **H**; *Multi?* adds a prefix, m_-, to the *Rel_Pair* features, if the **T**-path is longer than one dependency relation; *Dep_Same?* checks whether the two dependency types are the same, i.e., syntactic and semantic dependencies; *Rel_Sim?* only occurs when two semantic dependencies are compared, indicating whether they have the same prefixes, e.g., *C-*, *AM-*, etc.; *Rel_Same?* checks whether the two dependency relations are the same; and *Rel_Pair* simply concatenates the two relation labels together. Notice that the first seven feature types all contain boolean values, and we make the last one

boolean by observing whether that pair of dependency labels appears or not.

9.2.2 TSR Recognition

Being similar to entailment recognition, the TSR recognition is also based on a two-stage classification. Table 9.2 compares the two systems.

	the RTE system (Chapter 8)	→	the TSR system (this chapter)
the 1st stage	*relatedness* (and UNKNOWN)	→	*relatedness* ***inconsistency*** ***inequality***
the 2nd stage	ENTAILMENT and CONTRADICTION	→	PARAPHRASE ENTAILMENT CONTRADICTION UNKNOWN

Table 9.2: Comparison of the RTE system and the TSR system

At the first stage, we obtain all the features mentioned above and train three classifiers for the three measurements, *relatedness*, *inconsistency*, and *inequality*, and test on the whole dataset to obtain the numerical values. Table 9.3 shows the training material for each classifier. After that, we use these three measurements as the input features for the second stage classification, which gives us the final result.

	relatedness	*inconsistency*	*inequality*
PARAPHRASE	+	−	−
ENTAILMENT	+	−	+
CONTRADICTION	+	+	+
UNKNOWN	−	−	+

Table 9.3: Training data of the three classifiers

Notice that currently we have not done any feature selection. Instead, we mainly leave it to the SVM-based training to assign different weights to the features. But a careful feature engineering is definitely a worthy direction to work on, which will be left for the future.

9.3   Experiments

In this section, we present several experiments to evaluate our approach. We firstly describe the datasets we used, the details of which have been described in Chapter 7. Then we explain the preprocessing and configurations of our system. In the end, the results are shown, followed by an analysis of the data and further discussions.

9.3.1   Datasets

In Section 7.1, we have already described the datasets and corpora construction. Here, Table 9.4 shows an overview of the datasets and annotation we use in our experiments and we briefly repeat them in the following as a reminder.

Corpora	Paraphrase (P)	Entailment (E)	Contradiction (C)	Unknown (U)
AMT (584)		Facts (406)	Counter-Facts (178)	
MSR (5841)	Paraphrase (3940)	Non-Paraphrase (1901)		
PETE (367)		YES (194)	NO (173)	
RTE (2200)	ENTAILMENT (1100)		CONTRADICTION (330)	UNKNOWN (770)
TSR (260)	Equality (3)	Forward/Backward Entailment (10/27)	Contradiction (17)	Overlapping &Independent (203)
Total (9252)	3943	637	525	973

Table 9.4: Collection of heterogenous datasets with different annotation schemes, with the number of **T-H** pairs.

AMT   is a dataset we constructed using the crowd-sourcing technique (Wang and Callison-Burch, 2010). We used Amazon's Mechanical Turk[2], online non-expert annotators (Snow et al., 2008) to perform the task. Basically, we show the Turkers a paragraph of text with one highlighted named-entity and ask them to write some facts or counter-facts about it. There are three blank lines given for the annotators to fill in. For

---
[2]https://www.mturk.com/mturk/

each task, we show five texts, and for each text, we ask three Turkers to accomplish the task it. In all, we have collected 406 valid facts and 178 counter-facts, which are viewed as ENTAILMENT and CONTRADICTION respectively (more details can be found in Section 7.3).

MSR is a paraphrase corpus provided by Microsoft Research (Dolan and Brockett, 2005). It is a collection of manually annotated sentential paraphrases. This dataset consists of 5841 pairs of sentences which have been extracted from news sources on the web, along with human annotations indicating whether each pair captures a paraphrase/semantic equivalence relationship.

PETE is taken from the SemEval-2010 Task #12, Parser Evaluation using Textual Entailment[3] (Yuret et al., 2010). The dataset contains 367 pairs of texts in all and has a focus on entailments involving mainly the syntactic information. The annotation is two-way, YES is converted into ENTAILMENT and NO can be either CONTRADICTION or UNKNOWN. Since each text pair only concerns one syntactic phenomenon, the entailment relation is directional, excluding the paraphrases.

RTE is a mixture of RTE-4 (1000) and RTE-5 (1200) datasets. Both have three-way annotations, but the ENTAILMENT cases actually include PARAPHRASE as well. We did not include the unofficial three-way annotation of the RTE-3 pilot task.

TSR is the dataset we annotated under the annotation scheme mentioned in Section 7.2. The sentence pairs were extracted from the the RST Discourse Treebank (RST-DT)[4]. The annotation was done by two annotators in two rounds. The inter-annotator agreement is 91.2% and the kappa score is 0.775. We take all the valid and agreed sentence pairs (260) as the TSR dataset here (more details can be found in Section 7.2).

We consider the unidirectional relations between an ordered pair of texts (i.e., from the first one (**T**) to the second one (**H**)), *forward entailment* and *backward entailment* can be collapsed into one. We still use the name ENTAILMENT, but we strictly mean a directional relation, i.e.,

---

[3]http://pete.yuret.com/guide

[4]Available from the LDC: http://www.ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2002T07

T entails **H**, but **H** does not entail **T**. The original goal of having both *overlapping* and *independent* is to capture the spectrum of relatedness. However, in practice, even the human annotators found it difficult to agree on many cases. Therefore, we also collapse the last two relations into one, UNKNOWN, following the RTE label convention. *Equality* is the same as PARAPHRASE.

We randomly sample 250 **T-H** pairs from each dataset as the test sets (1000 pairs in all). The rest of the data are then randomly selected to create a balanced training set with an equal number of instances (i.e., text pairs) from each class.

9.3.2   Preprocessing

Within the scope of this chapter, we generally refer to all the linguistic analyses on the texts before feature extraction as *preprocessing*. The output of this procedure is a unified graph representation, which approximates the meaning of the input text. In particular, after tokenization and POS tagging, we conduct dependency parsing and semantic role labeling.

Tokenization and POS Tagging   We use the Penn Treebank style tokenization throughout the various processing stages. **TnT**[5] (Brants, 2000), an HMM-based POS tagger trained with Wall Street Journal sections of the PTB, was used to automatically predict the part-of-speech of each token in the texts and hypotheses.

Dependency Parsing   For obtaining the syntactic dependencies, we use two dependency parsers, MSTParser (McDonald et al., 2005) and MaltParser (Nivre et al., 2007). MSTParser is a graph-based dependency parser where the best parse tree is acquired by searching for a spanning tree which maximizes the score on an either partially or fully connected dependency graph. MaltParser is a transition-based incremental dependency parser, which is language-independent and data-driven. It contains a deterministic algorithm, which can be viewed as a variant of the basic shift-reduce algorithm. The combination of two parsers achieves state-of-the-art performance. Figure 9.1 shows the resulting syntactic dependency trees of the following **T-H** pair.

---

[5]`http://www.coli.uni-saarland.de/~thorsten/tnt/`

Semantic Role Labeling The statistical dependency parsers provide shallow syntactic analyses of the entailment pairs through the limited vocabulary of the dependency relations. In our case, the CoNLL shared task dataset from 2008 was used to train the statistical dependency parsing models. While such dependencies capture interesting syntactic relations, the contained information is not as detailed, when compared to the parsing systems with deeper representations. To compensate for this, we used a shallow semantic parser to predict the semantic role relations in the **T** and **H** of entailment pairs. The shallow semantic parser was also trained with the CoNLL 2008 shared task dataset, with semantic roles extracted from the Propbank and Nombank annotations (Surdeanu et al., 2008). Figure 9.2 shows the resulting semantic dependency graphs of the **T**-**H** pair.

9.3.3 Configurations and Results

For comparison, we configure our system in the following two ways to compose different baseline systems:

1. From the **classification strategy** perspective, the direct four-class classification is the baseline (*Direct Joint* in Table 9.5), compared with the main system with a two-stage classification (*3-D Model*);

2. From the **feature set** point of view, we take the bag-of-words similarity as the baseline[6] (*Direct BoW*), compared with the main system using both syntactic and semantic dependency structures (i.e., the *3-D Model*).

For the shortest path algorithm, we use the jGraphT package[7]. For the parameters of the joint dependency graph, we assign 0.5 for the semantic dependencies and 1.0 to all the syntactic dependencies, in order to give prior to the former when both exist; and for the machine-learning-based classifier, we use the UniverSVM package[8].

Table 9.5 shows the accuracy of the system performance. *Direct BoW* means the direct 4-class classification using bag-of-words similarity; *Direct Joint* uses the feature model based on dependency paths of the joint

---

[6]The bag-of-words similarity has been shown to be a strong baseline for RTE in the previous challenges.

[7]http://jgrapht.sourceforge.net/

[8]http://www.kyb.mpg.de/bs/people/fabee/universvm.html

Systems	4-Way (C, E, P, U)	3-Way (C, E&P, U)	2-Way (E&P, Others)	(P, Others)
Direct BoW	39.3%	54.5%	63.2%	62.1%
Direct Joint	42.3%	50.9%	66.8%	77.3%
3-D Model	45.9%	58.2%	69.9%	79.6%

Table 9.5: Results of the system with different configurations and different evaluation metrics.

graph and performs a direct classification as well; *3-D Model* builds three classifiers first and then builds another classifier based on the three values. *3-Way* follows the three-way RTE annotation scheme; the two *2-Way* annotations are two-way RTE and paraphrase identification respectively.

Notice that E here indicates the strict directional entailment excluding the bidirectional ones (i.e., P), which makes the task much harder (as we see more in Section 9.3.4). Nevertheless, the main approach, 3-D Model, improves the system performance greatly in all aspects, compared with the baselines. Apart from the self-evaluation, we also compare our approach with others' systems. Due to the difference in dataset, the numbers are only indicative.

RTE	3-Way (C, E&P, U)	2-Way		
		Acc.	Prec.	Rec.
3-D Model	58.2%	**69.9%**	**75.9%**	53.4%
*MacCartney and Manning (2007)	-	59.4%	70.1%	36.1%
*Heilman and Smith (2010)	-	62.8%	61.9%	**71.2%**
Our Prev.	**59.1%**	69.2%	-	-
*RTE-4 Median	50.7%	61.6%	-	-
*RTE-5 Avg.	52.0%	61.2%	-	-

Table 9.6: System comparison under the RTE annotation schemes

For the RTE comparison (Table 9.6[9]), the datasets are partially different due to the mixture of datasets. For reference, we re-run our previous system on the new dataset (indicated as *Our Prev.*, which was one of the top system in the previous RTE challenges). The results show that our new approach (*3-D Model*) is comparable to the previous system on the three-way RTE and outperforms it greatly on the two-way task. And both systems achieve much better results than the average. The system

---

[9]Asterisk indicates the different datasets.

based on natural logic (MacCartney and Manning, 2007) is precision-oriented while the system described in (Heilman and Smith, 2010) is recall-oriented. Our system achieves the highest precision among them.

P vs. Non-P	Acc.	Prec	Rec.
3-D Model	**79.6%**	57.2%	72.8%
*Das and Smith (2009) (QG)	73.9%	74.9%	**91.3%**
*Das and Smith (2009) (PoE)	76.1%	**79.6%**	86%
*Heilman and Smith (2010)	73.2%	75.7%	87.8%

Table 9.7: System comparison under the paraphrase identification task

Besides the RTE task, we also compare our approach with other paraphrase identification systems (Table 9.7[10]). Das and Smith (2009) proposed two systems, one with high-recall (QG, using a quasi-synchronous grammar) and the other with high-precision (PoE, using a product of experts to combine the QG model with lexical overlap features). Heilman and Smith (2010) refers to the same system as in Table 9.6. Although our system has lower precision and recall, our accuracy ranks the top, which indicates that our approach is better at non-paraphrase recognition.

Notice that our system is not fine-tuned to any specific recognition task. Instead, we built a general framework for recognizing all four TSRs. We also include heterogenous datasets collected by various methods in order to achieve the robustness of the system. However, if one is interested in recognizing one specific relation, a closer look at the data distribution can help with the feature selection.

9.3.4 Discussion

While the empirical results show a practical advantage of applying the three-dimensional space model to the TSR recognition task, in this subsection, we investigate whether this simplified semantic relation space with the chosen axises is a good approximation for these TSRs.

We plot all the test data into this space. Figure 9.4, Figure 9.5, and Figure 9.6 shows three different projections onto each two-dimensional plane.

Although the improvement on recognition accuracy is encouraging, these three measurements cannot fully separate different TSRs in this

---

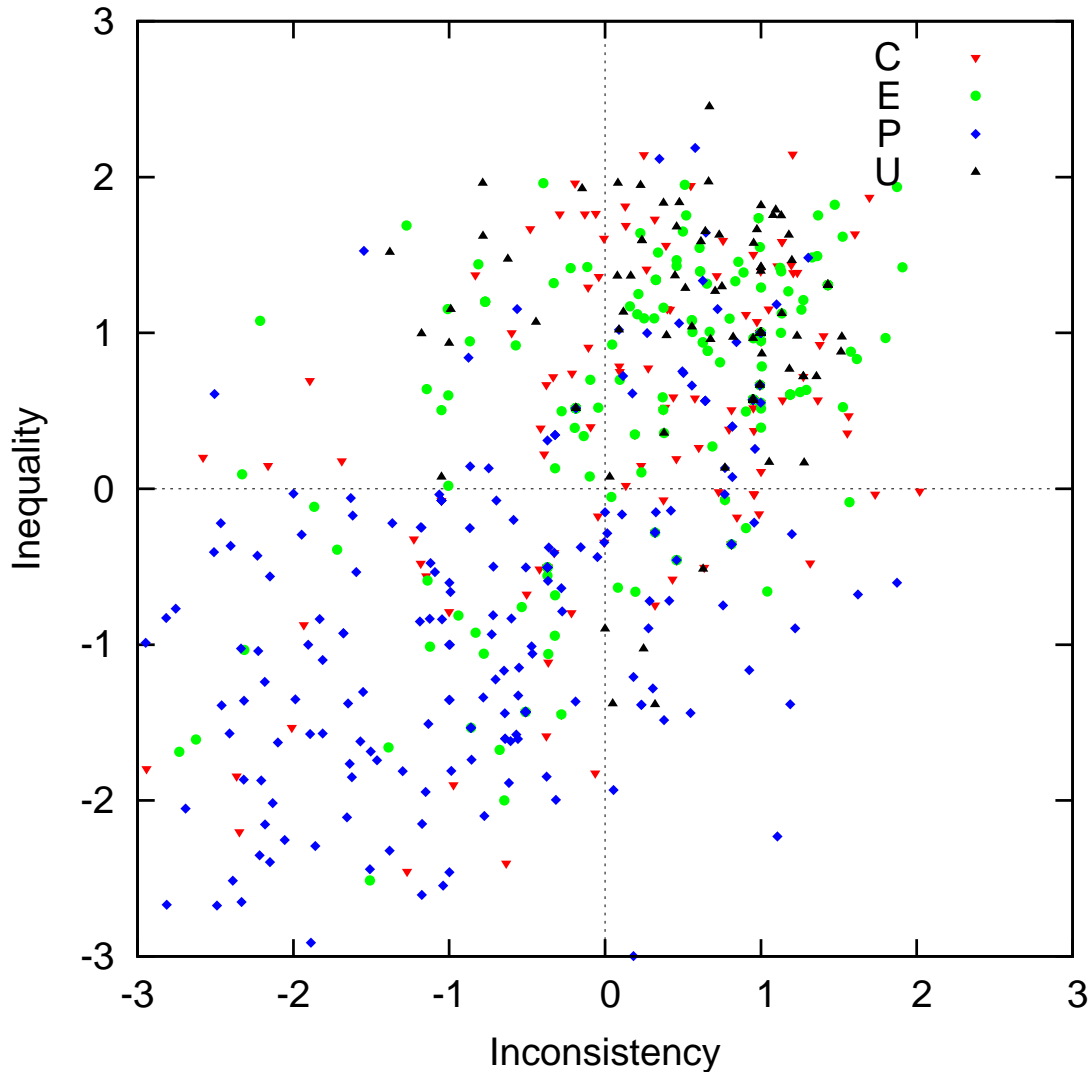[10]Asterisk indicates the different test sets.

Figure 9.4: Test data in the three-dimensional semantic relation space projected onto the three planes.

space. *P* clearly differs from the others and most of the data points stay in the region of low inconsistency (i.e., consistent), low inequality (i.e., equal), and high relatedness. However, the other three TSRs behave rather similarly to each other in terms of the regions.

Figure 9.7, Figure 9.8, and Figure 9.9 shows the other three TSRs on the same plane, *inconsistency-inequality*. Although the general trend of these three groups of data points is similar, slight differences do exist. *U* is rather restricted in the region of high inconsistency and high inequality; while the other two spread a bit over the whole plane. We did expect
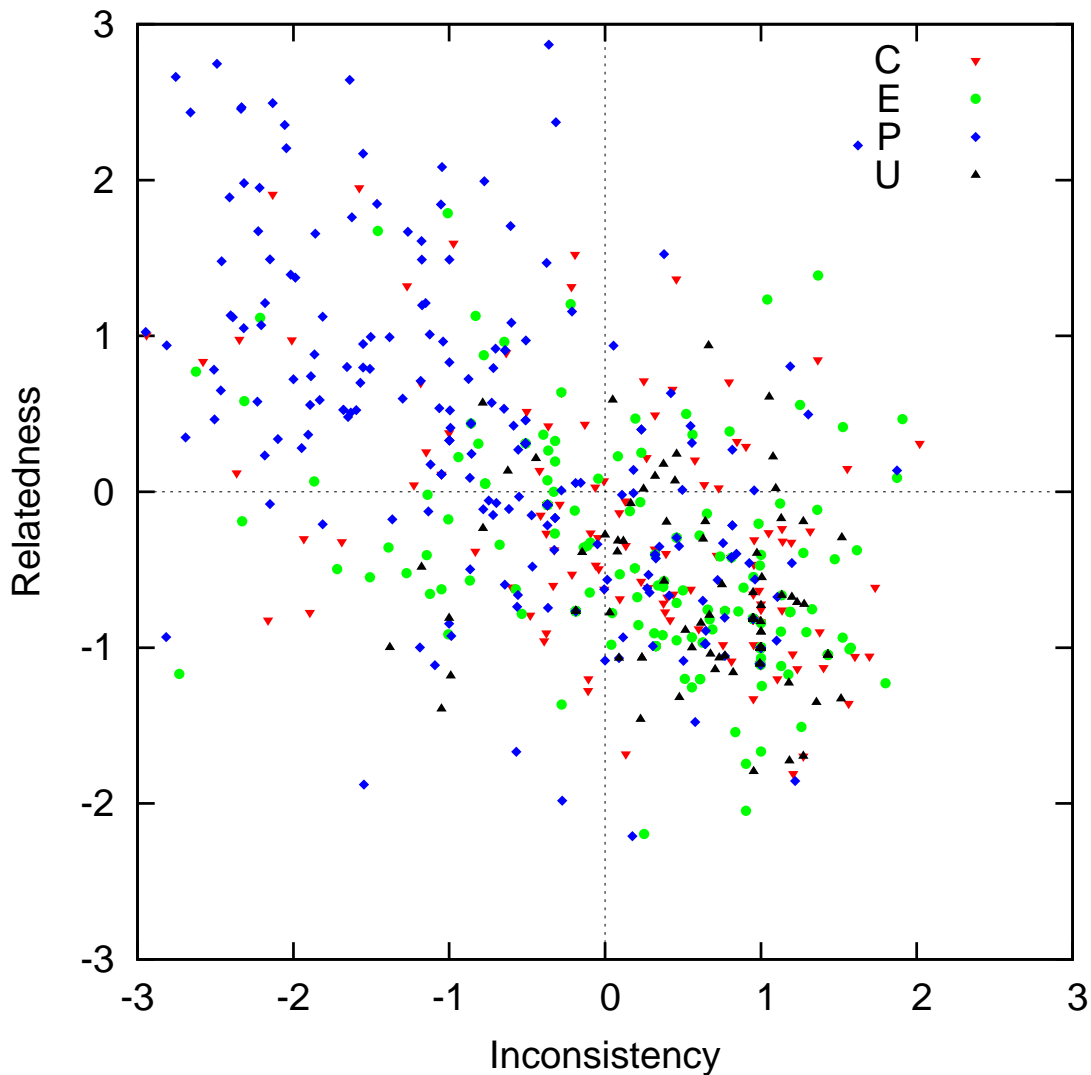
Figure 9.5: Test data in the three-dimensional semantic relation space projected onto the three planes.

the contrary behavior of C and E in terms of inconsistency, but it seems that our inconsistency measuring module is not as solid as the relatedness measure. This is in accordance with the fact that for the original three-way RTE task C is also the most difficult category to be recognized.

An even more difficult measurement is the inequality. Among all the four TSRs, the worst result is on E, which roots from the suboptimal inequality recognition. In retrospect, the matching methods applied to the **T-H** pair cannot capture the directionality or the semantic implication, but rather obtain a symmetric measurement, and this symmetry also ex-
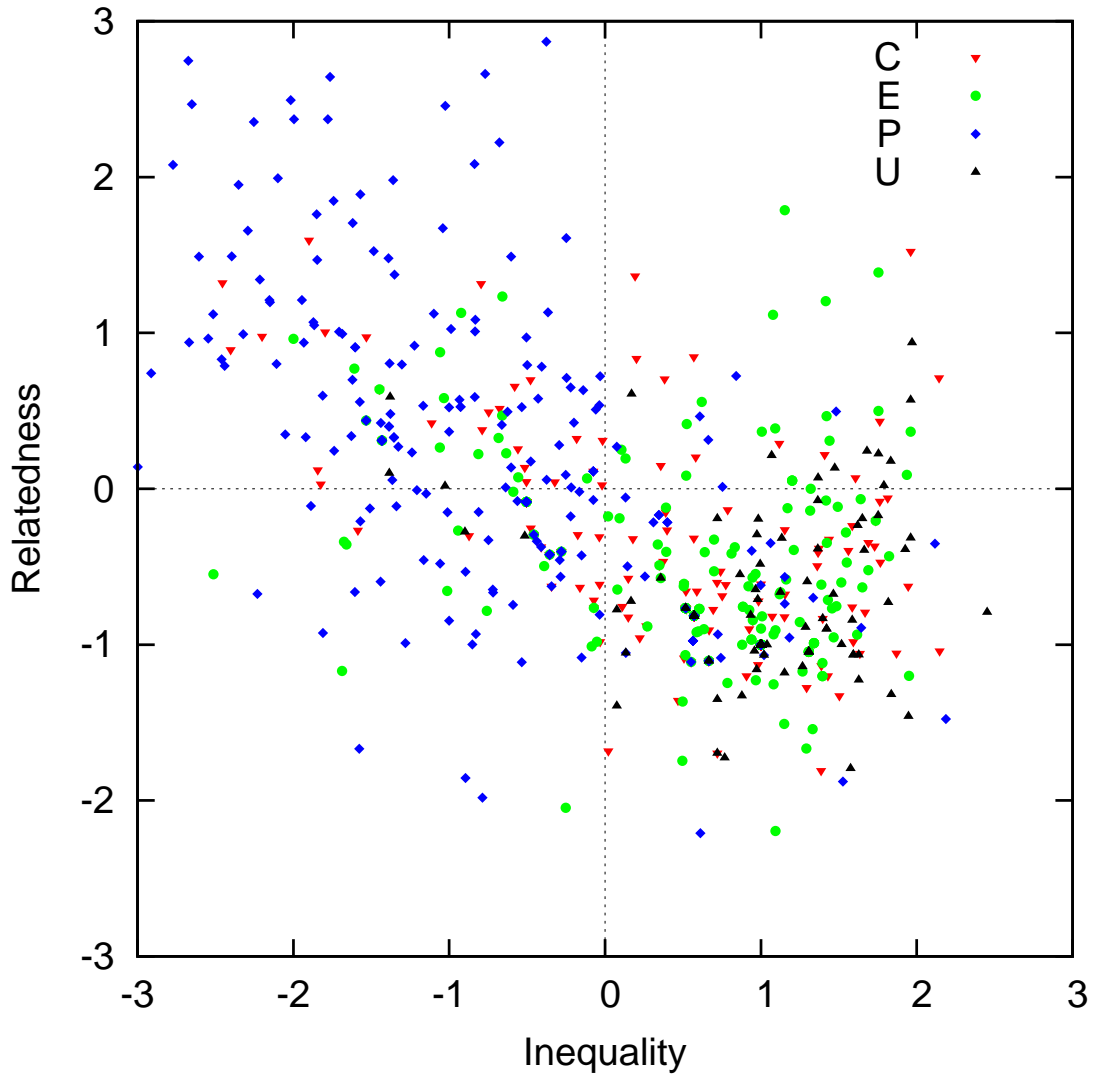
Figure 9.6: Test data in the three-dimensional semantic relation space projected onto the three planes.

plains the success of paraphrase recognition. Additionally, this may also suggest that, in the traditional RTE task, the high performance may attribute to the *P section* of the entailment, while the real directional *E* is still very difficult to catch.
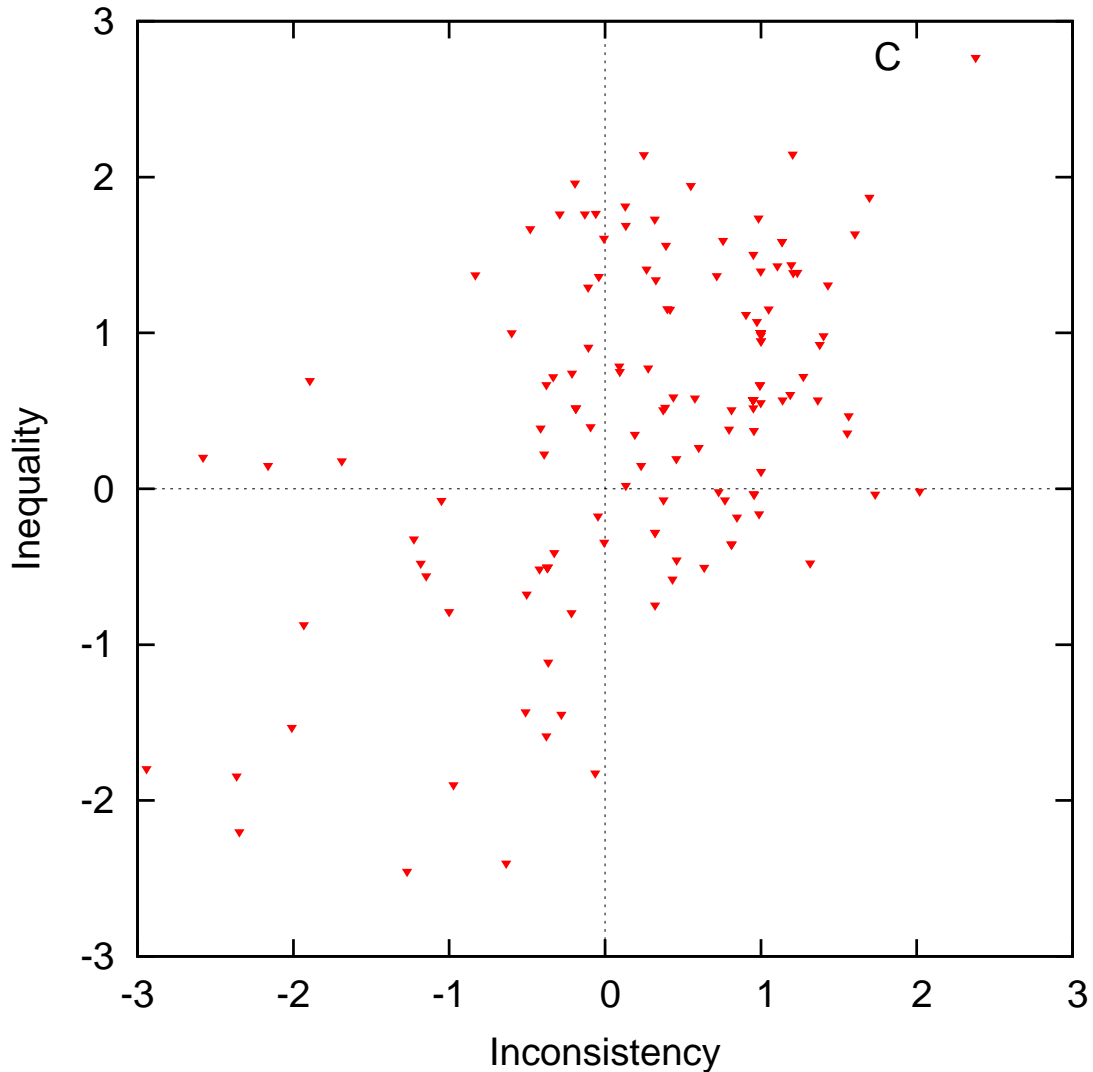
Figure 9.7: C, E, and U test data projected onto the inconsistency-inequality plane.

9.4 Summary and Future Extensions

In this chapter, we firstly show the generalization of the meaning representation based on dependency structures. Then, we present our approach of recognizing different textual semantic relations based on one three-dimensional model. *Relatedness*, *inconsistency*, and *inequality* are considered as the basic measurements for the recognition task, which are also the dimensions of the semantic relation space. We show empirically the effectiveness of this approach with a feature model based on depen-
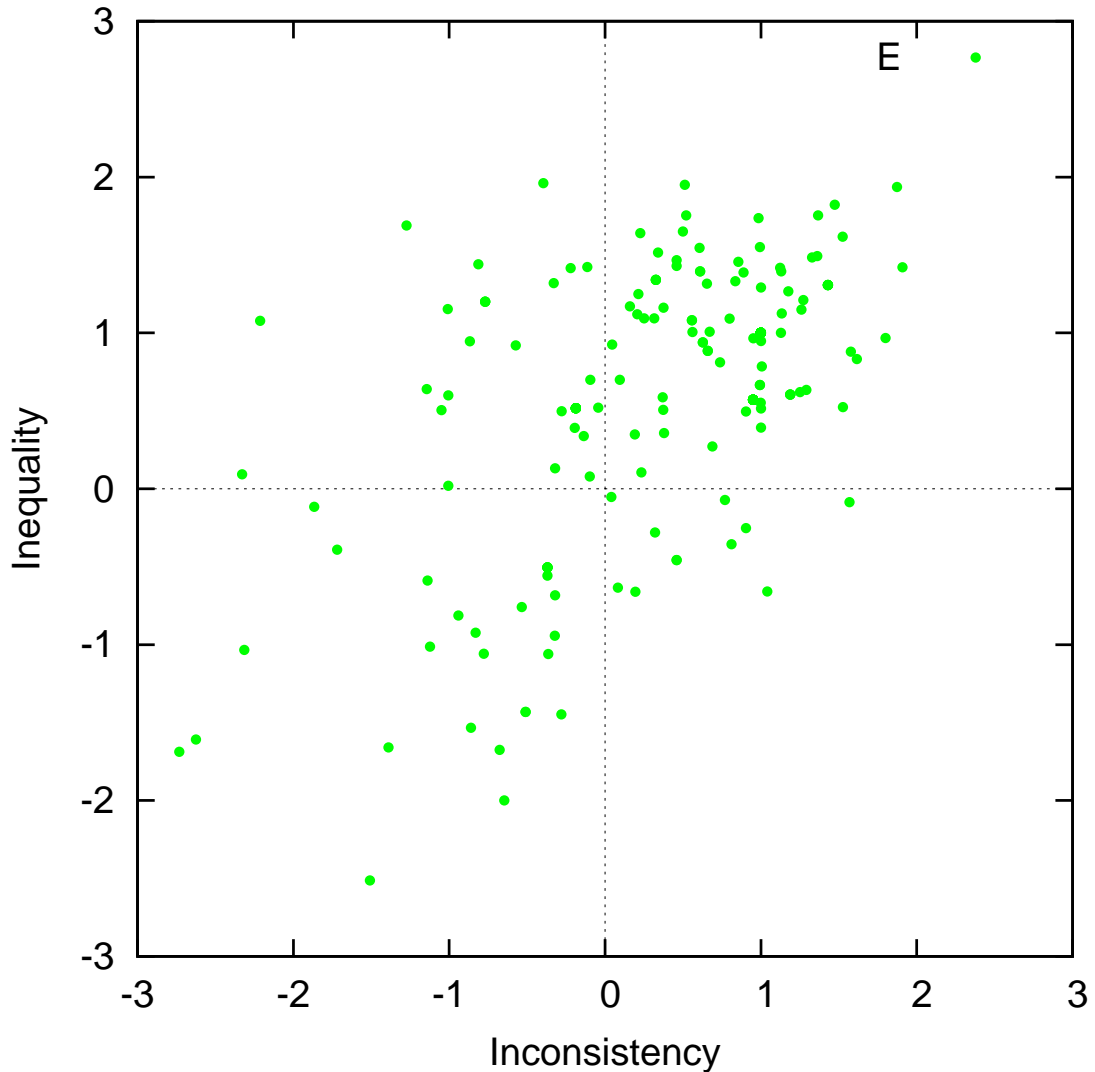
Figure 9.8: C, E, and U test data projected onto the inconsistency-inequality plane.

dency paths of the joint syntactic and semantic graph. We also interpret the results and the remaining difficulties visually.

There are three aspects we can improve the approach:

1. Inequality seems to be difficult to define and to measure, which suggests to consider other possible dimensions.

2. We are looking for a systematic way to tune the general system for specific TSR recognition tasks.

3. We have not incorporated lexical resources (e.g., WordNet) into our
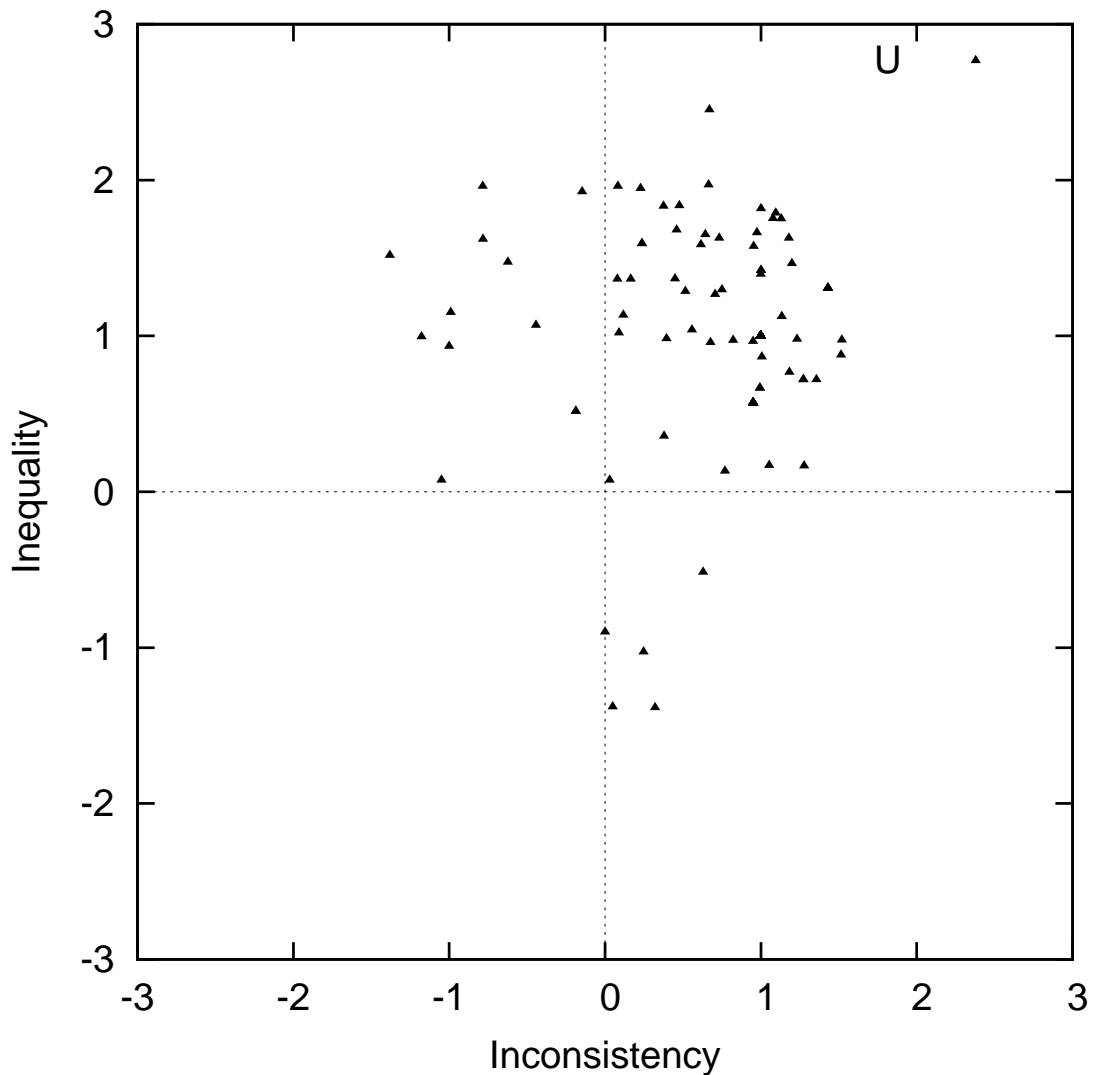
Figure 9.9: C, E, and U test data projected onto the inconsistency-inequality plane.

system yet, for a proper way of integration is still up for future research.

There are other extensions we want to make in the future as well:

- The meaning representation can be further enriched with other information, such as named-entities and their relations, or even deeper semantic relations like the scope of quantifiers.

- Enlightened by the specialized RTE modules we developed before (Chapter 3), we can also design specialized modules for other TSRs.

We provide more discussions in Chapter 10.

10 Summary and Perspectives

Many interesting issues arise during the dissertation time. In this chapter, only some of them are covered. In each of the following sections, we start with a summary of what we have done and then discuss on possible extensions. This includes candidate extensions to the current architecture for RTE, the possible improvement for other TSR recognition tasks, and several applications of the RTE system as a valuable component. We also provide several perspectives on the future exploration.

10.1   Intrinsic Approaches

In the first part of this dissertation, an extensible architecture is presented in Chapter 3, which consists of specialized RTE modules. Each RTE module is responsible for one specific type of entailment, aiming at a subset of the data in practice. Compared with the traditional pipeline systems, our system prefers precision-oriented modules to recall-oriented ones. Inside each module, one submodule selects the candidate text pairs to process, and another submodule decides whether there existed an entailment relation between them.

Chapter 4 shows one specialized RTE module, which focuses on those text pairs containing temporal expressions. Temporal expressions are used as anchors on the dependency trees to find the corresponding events contained in both texts (the first submodule), and then rules are applied to determine whether the entailment holds for that text pair (the second submodule). Chapter 5 also describes another kind of specialized RTE module, which depends on a textual inference rule collection, i.e., DIRT. The target of this module is the subset of the data to which some textual inference rule can be applied (the first submodule). Once at least one rule can be applied, the entailment holds for that text pair (the second submodule).

Experiments indicates that although both modules can only cover a small portion of the whole dataset, on those text pairs covered, they largely outperforms the baselines. In addition, the modules lead to a significant improvement on the entire dataset. The experiments also show that the extension of the TACTE system into other types of named-entities is not so successful. Although the coverage is promising (almost half of the dataset), the accuracy dropped to the baseline level. This confirms the high precision requirement for such specialized modules.

The natural extension of the current approach is to add more specialized modules. Apart from the modules dealing with entailment containing temporal expressions, or other named-entities, and those text pairs covered by the DIRT rules, many other modules can be considered:

- We can add a specialized module dealing with entailments containing negations. The candidate selector just selects those text pairs containing negation words, and the entailment detector changes the polarity of the result decided by other parts of the text. One challenging issue is to determine whether the identified negation words

are actually relevant to the entailment decision or just irrelevant information.

- A further extension of the previous module is to consider modal verbs. Since modal verbs are closed-class words, we can collect them and group them into several categories, *factual, counter-factual*, etc. The entailment detector is more complicated, since the polarity of the text can be determined by the combination of modality and negation, let alone the scopes of these words (Nairn et al., 2006).

- We may also consider a theorem prover based on logical forms to deal with quantifiers. The candidate selector chooses those text pairs which contain quantifiers, and the entailment detector obtains the answer through deductive reasoning.

- Once we have another external resource, we can easily build up a module based on it. For instance, if we have a paraphrase collection, we can apply them to the task of recognizing bi-directional entailment. If we have a gazetteer of location names in Europe, we can either improve the named-entity recognition part or build a standalone module only to cover those cases containing the names in the gazetteer.

Another aspect for improvement is the voting strategy. So far, we have not investigated much in the combination of all the outputs from the RTE modules, but just a simple voting based on the performance of the modules on the development dataset. It is worth looking at more fine-grained ranking approaches for this subtask. In particular, the question of how to resolve conflicts between the decisions of different modules needs further exploration. In addition, depending on the different voting strategy, the choice between precision- and recall-oriented approaches is (again) interesting to investigate.

Furthermore, the modules can be more interactive with each other, as well as with the preprocessors. Just as named-entity recognition and dependency parsing can benefit from each other's outputs, the RTE module dealing with named-entity resolution can also help the external inference rule application and vice versa. Complex cases of entailment do need such "collaboration" between modules.

10.2   Extrinsic Approaches

In the second part of the dissertation, we consider the relationship between textual entailment and other semantic relations between texts. Chapter 6 presents a generalization of the RTE task, which leads to a classification of four relations, PARAPHRASE, ENTAILMENT, CONTRADICTION, and UNKNOWN. Then three numerical features are proposed to characterize them, *relatedness, inconsistency,* and *inequality.*

Before doing the classification of textual semantic relations, the corpora construction is introduced in Chapter 7. An overview of several existing corpora is given, as well as a discussion of the methodologies used during the construction. Then the work on constructing two alternative corpora for textual semantic relations is presented, one constructed by manual annotation and the other using a crowd-sourcing technique to collect data from the Web. Based on inter-annotator agreement and analysis of the sampled data, both corpora show comparable quality to other existing corpora. These corpora are all used as datasets in our experiments.

Chapter 8 describes the approach of using relatedness recognition as an intermediate step for entailment recognition. The evaluation confirms that the two-stage classification method works better than three-way classification on the RTE data. We further extend the system with two other measurements, inconsistency and inequality, and use them to classify multiple semantic relations (Chapter 9). The results show that not only one single recognition task (i.e., RTE) can benefit from the search space reduction, but also multiple tasks can be accomplished in one unified framework, like paraphrase acquisition and contradiction detection.

Among the three features we considered for the TSR classification, inconsistency and inequality are difficult to measure. For the former measurement, after adding the modules dealing with negation and modal verbs, the performance can be improved. For the latter, we can compare several lexical resources as we did for relatedness recognition, although the directionality between words is also not trivial to obtain (Kotlerman et al., 2009).

We can also do detailed feature engineering for acquiring these three measurements. For instance, synonyms and antonyms are important to *relatedness,* but probably do not have any impact on *inequality,* since they are both bi-directional relations. *Inconsistency* can be detected when one contradictory part is discovered, while *relatedness* has to go through all

the information contained in the text pair. These suggest that we should take different approaches to acquire different measurements.

In addition, for entailment recognition, we currently use *intersection* to combine all the results from comparing two semantic units. However, that does not always provide the best result. The *union* operator is also interesting to explore, since we aim at identifying all possible semantic relations between the two texts.

In fact, the *semantic unit* itself (i.e., the meaning representation) can be extended to incorporate more information. For instance, the named-entities and their relations can also be represented in the dependency style, or even the scope information of the quantifier in formal semantics. Furthermore, if we properly combine the results from research on lexical semantics with our current architecture, the monotonicity issue may also be systematically handled.

An even more attractive method is to integrate the intrinsic approaches with the extrinsic ones. For each TSR recognition, we may have several specialized modules. Each specialized module can select a subset of the data and deal with it. The external knowledge resources can also aim at different subsets. Accordingly, the voting strategy needs to be "upgraded" to handle conflicts between different semantic relation decisions.

10.3　Applications

We have discussed the motivation for tackling the textual entailment problem at the beginning of this dissertation, but we have not elaborated on using the system as a component for other downstream applications[1]. Here, we briefly introduce three tasks, *answer validation*, *relation validation*, and *parser evaluation*, where we use previously developed RTE systems as valuable components to tackle the problems.

Answer Validation   is a task proposed by the Cross Language Evaluation Forum (CLEF) (Peñas et al., 2007, Rodrigo et al., 2008) and it aims at developing systems able to decide whether the answer of a question answering system is correct or not. The input is a set of pairs <*answer*, *supporting text*> grouped by *question*. Participant systems must return

---

[1]We described several work of applying the existing RTE system to other NLP tasks in Section 2.6.

one of the following values for each answer: validated, selected, and rejected. The first and the last ones are straightforward, and the second one marks the best answer when there is more than one correct answer to a question.

Our system uses the RTE module (Section 5.4.2) as a core component. We adapt questions, their corresponding answers, and supporting documents into **T-H** pairs, assisted by some manually designed patterns. Then, the task can be cast as an entailment recognition task. The answer is correct when the entailment relation holds and vice versa. We achieved the best results for both English and German languages in the evaluation[2] (Wang and Neumann, 2008a).

Relation Validation can be described as follows: given an instance of a relation between named-entities and a relevant text fragment, the system is asked to decide whether this instance is true or not. We also made use of the RTE module (Section 5.4.2) as the core component and transformed the task into an RTE problem, meaning that the relation is validated when the entailment holds and vice versa. We set up two different experiments to test our system: one is based on an annotated data set; the other is based on real web data via the integration of our system with an existing information extraction system. The results suggest that recognizing textual entailment is a feasible way to address the relation validation task as well (Wang and Neumann, 2008b).

Parser Evaluation using Textual Entailment (PETE) (Yuret et al., 2010) is the SemEval-2010 Task[3] #12, which is an interesting task connecting two areas of research, parsing and RTE. The former is usually concerned with syntactic analysis in specific linguistic frameworks, while the latter is believed to involve more semantic aspects of the language, although in fact no clear-cut boundary can be drawn between syntax and semantics for both tasks. The basic idea is to evaluate (different) parser outputs by applying them to the RTE task. The advantage is that this evaluation scheme is formalism independent (for the parsers).

The RTE module used in our participating system is mainly described in Chapter 9. Instead of using the 3-D model for TSR recognition (Section 9.2.2), we directly recognize the entailment relation based on the

---

[2]For the German language, we applied a German dependency parser (Neumann and Piskorski, 2002) for the preprocessing.

[3]`http://semeval2.fbk.eu/semeval2.php`

features (Section 9.2.1) extracted from the joint dependency graph (Section 9.1).   The best setting of our system ranks the 3rd place in the evaluation and different parsers behave differently in terms of both the parsing outputs and the final RTE accuracy (Wang and Zhang, 2010).

Although the evaluation on the downstream applications cannot faithfully reflect the performance of the RTE module, they suggest the directions for further improvements.   After all, we need to find a balance between ideal modules and practical solutions to the applications.

There is still a distance between these tasks and the real-life NLP applications. However, recognizing textual entailment and other textual semantic relations provide a generic way of comparing two given texts and capturing the relation between their meanings.   How to make full use of the outputs is still an open issue, but we look forward to more applications using RTE as core components in the future.

Bibliography

Rodrigo Agerri. Metaphor in textual entailment. In *Proceedings of COLING 2008*, Manchester, UK, 2008.

Eugene Agichtein, Walt Askew, and Yandong Liu. Combining Lexical, Syntactic, and Semantic Evidence for Textual Entailment Classification. In *Proceedings of the First Text Analysis Conference (TAC 2008)*, 2009.

Roni Ben Aharon, Idan Szpektor, and Ido Dagan. Generating entailment rules from framenet. In *Proceedings of ACL 2010 Conference Short Papers*, pages 241–246, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26:832–843, 1983.

A.R. Anderson and Jr. N.D. Belnap. *Entailment: The Logic of Relevance and Necessity*, volume I. Princeton University Press, Princeton, 1975.

A.R. Anderson, Jr. N.D. Belnap, and J.M. Dunn. *Entailment: The Logic of Relevance and Necessity*, volume II. Princeton University Press, Princeton, 1992.

I. Androutsopoulos and P. Malakasiotis. A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*, 38:135–187, 2010.

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The berkeley framenet project. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, pages 86–90, Montreal, Canada, 1998.

Alexandra Balahur, Elena Lloret, Óscar Ferrández, Andrés Montoyo, Manuel Palomar, and Rafael Muñoz. The dlsiuaes team's participation in the tac 2008 tracks. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Colin Bannard and Chris Callison-Burch. Paraphrasing with bilingual parallel corpora. In *Proceedings of ACL*, 2005.

R. Bar-Haim, I. Dagan, B. Dolan, L. Ferro, D. Giampiccolo, B. Magnini, and I. Szpektor. The second pascal recognising textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, 2006.

Roy Bar-Haim, Ido Dagan, Iddo Greental, Idan Szpektor, and Moshe Friedman. Semantic inference at the lexical-syntactic level for textual entailment recognition. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 131–136, Prague, June 2007. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W/W07/W07-1422`.

Roy Bar-Haim, Jonathan Berant, Ido Dagan, Iddo Greental, Shachar Mirkin, Eyal Shnarch, and Idan Szpektor. Efficient semantic deduction and approximate matching over compact parse forests. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

R. Barzilay and N. Elhadad. Sentence alignment for monolingual comparable corpora. In *Proceedings of EMNLP*, 2003.

Regina Barzilay and Lillian Lee. Learning to paraphrase: an unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 16–23, Edmonton, Canada, May 27-June 01 2003. Association for Computational Linguistics.

Regina Barzilay and Kathleen McKeown. Extracting paraphrases from a parallel corpus. In *Proceedings of ACL*, 2001. URL `http://www.cs.columbia.edu/nlp/papers/2001/barzilay_mckeown_01.pdf`.

Regina Barzilay, Kathleen McKeown, and Michael Elhadad. Information fusion in the context of multi-document summarization. In *Proceedings of ACL*, College Park, MD, 1999.

Roberto Basili, Diego De Cao, Paolo Marocco, and Marco Pennacchiotti. Learning selectional preferences for entailment or paraphrasing rules. In *In Proceedings of RANLP*, Borovets, Bulgaria, 2007.

Jeremy Bensley and Andrew Hickl. Workshop: Application of lcc's groundhog system for rte-4. In *Proceedings of the Text Analysis Confer-*

ence (TAC 2008) Workshop - RTE-4 Track, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

L. Bentivogli, B. Magnini, I. Dagan, H.T. Dang, and D. Giampiccolo. The fifth pascal recognizing textual entailment challenge. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Luisa Bentivogli, Elena Cabrio, Ido Dagan, Danilo Giampiccolo, Medea Lo Leggio, and Bernardo Magnini. Building textual entailment specialized data sets: a methodology for isolating linguistic phenomena relevant to inference. In *Proceedings of the 7th International Conference on Language Resources and Evaluation (LREC)*, Valletta, Malta, May 2010.

Jonathan Berant, Ido Dagan, and Jacob Goldberger. Global learning of focused entailment graphs. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1220–1229, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

Richard Bergmair. Monte carlo semantics: Mcpiet at rte4. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

L. Bergroth, H. Hakonen, and T. Raita. A survey of longest common subsequence algorithms. In *Proceedings of the Seventh International Symposium on String Processing and Information Retrieval*, pages 39–48, A Curuna , Spain, September 2000.

R. Bhagat, P. Pantel, and E. Hovy. Ledir: An unsupervised algorithm for learning directionality of inference rules. In *Proceedings of EMNLP-CoNLL*, 2007.

D. Bobrow, D. Crouch, T. King, C. Condoravdi, L. Karttunen, R. Nairn, V. de Paiva, and A. Zaenen. Precision-focused textual inference. In *Proceedings of the ACL Workshop on Textual Entailment and Paraphrasing*, Prague, Czech Republic, 2007.

J. Bos and K. Markert. Combining shallow and deep nlp methods for recognizing textual entailment. In *Proceedings of PASCAL Workshop on Recognizing Textual Entailment*, Southampton, UK, 2005.

Johan Bos and Katja Markert. When logical inference helps determining textual entailment (and when it doesn't). In *Proceedings of the Second PASCAL Challenges Workshop on Recognizing Textual Entailment*, 2006.

Johan Bos and Tetsushi Oka. A spoken language interface with a mobile robot. *Artificial Life and Robotics*, 11(1):42–47, January 2007.

Wauter Bosma and Chris Callison-Burch. Paraphrase substitution for recognizing textual entailment. paraphrase substitution for recognizing textual entailment. In C. Peters et al., editor, *Evaluation of Multilingual and Multimodal Information Retrieval*, Lecture Notes in Computer Science, 2007.

Thorsten Brants. TnT - a statistical part-of-speech tagger. In *Proceedings of the 6th Conference on Applied Natural Language Processing (ANLP 2000)*, Seattle, USA, 2000.

R. Bunescu and R. Mooney. Subsequence kernels for relation extraction. In *Advances in Neural Information Processing Systems 18*. MIT Press, 2006.

Aljoscha Burchardt, Nils Reiter, Stefan Thater, and Anette Frank. A semantic approach to textual entailment: System evaluation and task analysis. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, Prague, Czech Republic, 2007.

John Burger and Lisa Ferro. Generating an entailment corpus from news headlines. In *Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment*, pages 49–54, Ann Arbor, Michigan, USA, 2005. Association for Computational Linguistics.

Elena Cabrio, Milen Kouylekov, and Bernardo Magnini. Combining specialized entailment engines for rte-4. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Chris Callison-Burch. Syntactic constraints on paraphrases extracted from parallel corpora. In *Proceedings of EMNLP*, 2008.

Julio Javier Castillo and Laura Alonso i Alemany. An approach using named entities for recognizing textual entailment. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track,*

Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Asli Celikyilmaz and Marcus Thint. Semantic approach to textual entailment for question answering. In *IEEE International Conference on Cognitive Informatics*, Stanford University, CA, August 2008. IEEE CS Press.

Gennaro Chierchia and Sally McConnell-Ginet. *Meaning and Grammar: An Introduction to Semantics*. MIT Press, 2nd edition, March 2000.

Timothy Chklovski and Patrick Pantel. Verbocean: Mining the web for fine-grained semantic verb relations. In *Proceedings of EMNLP*, Barcelona, Spain, 2004.

Rudi Cilibrasi and Paul M. B. Vitanyi. The Google Similarity Distance. *IEEE/ACM Transactions on Knowledge and Data Engineering*, 19(3): 370–383, 2007.

Peter Clark and Phil Harrison. Recognizing Textual Entailment with Logical Inference. In *Proceedings of the First Text Analysis Conference (TAC 2008)*, Gaithersburg, Maryland, USA, November 2009a. National Institute of Standards and Technology.

Peter Clark and Phil Harrison. An inference-based approach to recognizing entailment. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009b. National Institute of Standards and Technology.

Peter Clark, Phil Harrison, John Thompson, William Murray, Jerry Hobbs, and Christiane Fellbaum. On the role of lexical and world knowledge in rte3. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 54–59, Prague, June 2007. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W/W07/W07-1409`.

Robin Cooper, Dick Crouch, Jan Van Eijck, Chris Fox, Johan Van Genabith, Jan Jaspars, Hans Kamp, David Milward, Manfred Pinkal, Massimo Poesio, and Steve Pulman. A framework for computational semantics (FraCaS). Technical report, The FraCaS Consortium, 1996.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In Quiñonero-Candela et al., editor,

MLCW 2005, volume LNAI Volume 3944, pages 177–190. Springer-Verlag, 2005.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In *Lecture Notes in Computer Science, Vol. 3944, Springer*, pages 177–190. Quiñonero-Candela, J.; Dagan, I.; Magnini, B.; d'Alché-Buc, F. Machine Learning Challenges, 2006. URL `http://www.aclweb.org/anthology/P/P08/P08-1118`.

Cristian Danescu-Niculescu-Mizil and Lillian Lee. Don't 'have a clue'? unsupervised co-learning of downward-entailing operators. In *Proceedings of ACL 2010 Conference Short Papers*, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

Cristian Danescu-Niculescu-Mizil, Lillian Lee, and Richard Ducott. Without a 'doubt'? unsupervised discovery of downward-entailing operators. without a 'doubt'? unsupervised discovery of downward-entailing operators. In *Proceedings of NAACL-HLT*, pages 137–145. Association for Computational Linguistics, 2009.

D. Das and N. A. Smith. Paraphrase identification as probabilistic quasi-synchronous recognition. In *Proceedings of ACL-IJCNLP 2009*, 2009.

Marie-Catherine de Marneffe, Bill MacCartney, Trond Grenager, Daniel Cer, Anna Rafferty, and Christopher D. Manning. Learning to distinguish valid textual entailments. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy, 2006.

Marie-Catherine de Marneffe, Anna N. Rafferty, and Christopher D. Manning. Finding contradictions in text. In *Proceedings of ACL-08: HLT*, pages 1039–1047, Columbus, Ohio, June 2008. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P/P08/P08-1118`.

E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

Georgiana Dinu and Rui Wang. Inference rules and their application to recognizing textual entailment. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL-09)*, pages 211–219, Athens, Greece, 2009. Association for Computational Linguistics.

Bill Dolan, Chris Quirk, and Chris Brockett. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of COLING*, 2004.

William B. Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the IWP2005*, 2005.

Witold Drozdzynski, Hans-Ulrich Krieger, Jakub Piskorski, Ulrich Schäfer, and Feiyu Xu. Shallow processing with unification and typed feature structures — foundations and applications. *Künstliche Intelligenz*, 1:17–23, 2004.

Christiane Fellbaum, editor. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press, May 1998. ISBN 026206197X. URL `http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20\&path=ASIN/026206197X`.

Óscar Ferrández, Rafael Munõz, and Manuel Palomar. Alicante university at tac 2009: Experiments in rte. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005)*, pages 363–370. Association for Computational Linguistics, 2005.

P. Fung and Y. Y. Lo. An IR approach for translating new words from nonparallel, comparable texts. In *Proceedings of ACL*, 1998.

Dimitrios Galanis and Prodromos Malakasiotis. Aueb at tac 2008. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Konstantina Garoufi. Towards a better understanding of applied textual entailment: Annotation and evaluation of the rte-2 dataset. Master's thesis, Saarland University, 2007.

Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*,

pages 1–9, Prague, June 2007. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W/W07/W07-1401`.

Danilo Giampiccolo, Hoa Trang Dang, Bernardog Magnini, Ido Dagan, Elena Cabrio, and Bill Dolan. The Fourth PASCAL Recognizing Textual Entailment Challenge. In *Proceedings of the First Text Analysis Conference (TAC 2008)*, 2009.

Demetrios G. Glinos. Recognizing textual entailment at rte4 with ceres. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

H. P. Grice. Logic and conversation. In P. Cole and J. L. Morgan, editors, *Synax and Semantics*, volume 3 (Speech Acts), pages 41–58. Academic Press, 1975.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the 13th Conference on Computational Natural Language Learning (CoNLL-2009)*, Boulder, CO, USA, 2009.

Benjamin Han, Donna Gates, and Lori Levin. From language to time: A temporal expression anchorer. *Proceedings of the Thirteenth International Symposium on Temporal Representation and Reasoning (TIME'06)*, pages 196 – 203, 2006.

Sanda Harabagiu and Andrew Hickl. Methods for using textual entailment in open-domain question answering. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 905–912, Sydney, Australia, July 2006. Association for Computational Linguistics. doi: 10.3115/1220175.1220289. URL `http://www.aclweb.org/anthology/P06-1114`.

Z. Harris. Distributional structure. In *Word*, 10(23), 1954.

Michael Heilman and Noah A. Smith. Tree edit models for recognizing textual entailments, paraphrases, and answers to questions. In *Proceedings of NAACL-HLT*, pages 1011–1019, 2010.

Andrew Hickl, John Williams, Jeremy Bensley, Kirk Roberts, Bryan Rink, and Ying Shi. Recognizing textual entailment with lcc's groundhog system. In *Proceedings of the Second PASCAL Challenges Workshop*, 2006.

Jerry R. Hobbs and Feng Pan. Time Ontology in OWL. W3C Working Draft 27 September 2006. http://www.w3.org/TR/2006/WD-owl-time-20060927/, 2006. URL `http://www.w3.org/TR/2006/WD-owl-time-20060927/`.

Richard Hudson. *Word Grammar*. Basil Blackwell Publishers Limited, Oxford, England, 1984.

Ali Ibrahim, Boris Katz, and Jimmy Lin. Extracting structural paraphrases from aligned monolingual corpora. In *Proceedings of ACL*, 2003.

Adrian Iftene. Uaic participation at rte4. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Adrian Iftene and Alexandra B. Dobrescu. Hypothesis transformation and semantic variability rules used in recognizing textual entailment. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 125–130. Association for Computational Linguistics, June 2007. URL `http://www.aclweb.org/anthology/W/W07/W07-1421`.

Adrian Iftene and Mihai-Alex Moruz. Uaic participation at rte5. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Karin Kipper, Anna Korhonen, Neville Ryant, and Martha Palmer. Extending verbnet with novel verb classes. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2006)*, Genoa, Italy, June 2006.

Dan Klein and Christopher D. Manning. Accurate unlexicalized parsing. In *Proceedings of the 41st Meeting of the Association for Computational Linguistics*, pages 423–430, 2003.

Donald Knuth. *Sorting and Searching*, volume 3 of *The Art of Computer Programming*, page 159. Addison-Wesley Professional, 2nd edition, 1998.

Lili Kotlerman, Ido Dagan, Idan Szpektor, and Maayan Zhitomirsky-Geffet. Directional distributional similarity for lexical expansion. In *Proceedings of ACL*, 2009.

Milen Kouylekov and Matteo Negri. An open-source package for recognizing textual entailment. In *Proceedings of the ACL 2010 System Demonstrations*, pages 42–47, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

Ralf Krestel, Sabine Bergler, and René Witte. A belief revision approach to textual entailment recognition. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009a. National Institute of Standards and Technology.

Ralf Krestel, René Witte, and Sabine Bergler. Believe it or not: Solving the tac 2009 textual entailment tasks through an artificial believer system. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009b. National Institute of Standards and Technology.

Klaus Krippendorff. *Content Analysis: An Introduction to Its Methodology.* Sage Publications, 1980.

David Lewis. *On the Plurality of Worlds.* Oxford and New York: Basil Blackwell, 1986.

Fangtao Li, Zhicheng Zheng, Fan Bu, Yang Tang, Xiaoyan Zhu, and Minlie Huang. Thu quanta at tac 2009 kbp and rte track. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009a. National Institute of Standards and Technology.

Fangtao Li, Zhicheng Zheng, Yang Tang, Fan Bu, Rong Ge, Xiaoyan Zhu, Xian Zhang, and Minlie Huang. Thu quanta at tac 2008 qa and rte track. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009b. National Institute of Standards and Technology.

Percy Liang, Ben Taskar, and Dan Klein. Alignment by agreement. In *Proceedings of NAACL*, 2006.

Dekang Lin. Dependency-based evaluation of minipar. In *Workshop on the Evaluation of Parsing Systems*, 1998.

Dekang Lin and Patrick Pantel. Dirt - discovery of inference rules from text. In *Proceedings of the ACM SIGKDD*, 2001.

Bill MacCartney and Christopher D. Manning. Natural logic for textual inference. In *Proceedings of the Workshop on Textual Entailment and Paraphrasing*, pages 193–200, Prague, Czech Republic, 2007. Association for Computational Linguistics.

Bill MacCartney, Michel Galley, and Christopher D. Manning. A phrase-based alignment model for natural language inference. In *Proceedings of EMNLP 2008*, 2008.

Prodromos Malakasiotis. Aueb at tac 2009. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Erwin Marsi, Emiel Krahmer, and Wauter Bosma. Dependency-based paraphrasing for recognizing textual entailment. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 83–88, Prague, 2007.

Yuval Marton, Chris Callison-Burch, and Philip Resnik. Improved statistical machine translation using monolingually-derived paraphrases. In *Proceedings of EMNLP*, Singapore, 2009.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajic. Non-Projective Dependency Parsing using Spanning Tree Algorithms. In *Proceedings of hlt-emnlp 2005*, pages 523–530, Vancouver, Canada, 2005.

Y. Mehdad and B. Magnini. A word overlap baseline for the recognizing textual entailment task. Online, 2009.

Yashar Mehdad, Alessandro Moschitti, and Fabio Massiomo Zanzotto. Semker: Syntactic/semantic kernels for recognizing textual entailment. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009a. National Institute of Standards and Technology.

Yashar Mehdad, Matteo Negri, Elena Cabrio, Milen Kouylekov, and Bernardo Magnini. Using lexical resources in a distance-based approach to rte. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009b. National Institute of Standards and Technology.

Yashar Mehdad, Matteo Negri, and Marcello Federico. Towards cross-lingual textual entailment. In *Proceedings of NAACL-HLT 2010*, pages 321–324, Los Angeles, California, USA, June 2010. Association for Computational Linguistics.

Shachar Mirkin, Ido Dagan, and Eyal Shnarch. Evaluating the inferential utility of lexical-semantic resources. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL-09)*, pages 558–566, Athens, Greece, March 2009a. Association for Computational Linguistics.

Shachar Mirkin, Lucia Specia, Nicola Cancedda, Ido Dagan, Marc Dymetman, and Idan Szpektor. Source-language entailment modeling for translating unknown terms. In *Proceedings of ACL-IJCNLP 2009*, Singapore, Singapore, 2009b. Association for Computational Linguistics.

Shachar Mirkin, Jonathan Berant, Ido Dagan, and Eyal Shnarch. Recognising entailment within discourse. In *Proceedings of COLING 2010*, Beijing, China, August 2010a.

Shachar Mirkin, Ido Dagan, and Sebastian Padó. Assessing the role of discourse references in entailment inference. In *Proceedings of ACL 2010*, Uppsala, Sweden, 2010b. Association for Computational Linguistics.

Dragos Stefan Munteanu and Daniel Marcu. Extracting parallel subsentential fragments from non-parallel corpora. In *Proceedings of ACL*, 2006.

Koji Murakami, Shouko Masuda, Suguru Matsuyoshi, Eric Nichols, Kentaro Inui, and Yuji Matsumoto. Annotating semantic relations combining facts and opinions. In *Proceedings of the Third Linguistic Annotation Workshop, ACL-IJCNLP-09*, pages 150–153, 2009.

Rowan Nairn, Cleo Condoravdi, and Lauri Karttunen. Computing relative polarity for textual inference. In *Proceedings of ICoS-5 (Inference in Computational Semantics*, Buxton, UK, 2006.

G. Neumann and J. Piskorski. A shallow text processing core engine. *Journal of Computational Intelligence*, 18(3):451–476, 2002.

Rodney D. Nielsen, Lee Becker, and Wayne Ward. Tac 2008 clear rte system report: Facet-based entailment. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Joakim Nivre, Jens Nilsson, Johan Hall, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(1):1–41, 2007.

Bahadorreza Ofoghi and John Yearwood. Ub.dmirg: A syntactic lexical system for recognizing textual entailments. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Sebastian Padó, Daniel Cer, Michel Galley, Daniel Jurafsky, and Christopher D. Manning. Measuring machine translation quality as semantic equivalence: A metric based on entailment features. *Machine Translation*, 23(2–3):181–193, 2009a.

Sebastian Padó, Marie-Catherine de Marneffe, Bill MacCartney, Anna N. Rafferty, Eric Yeh, and Christopher D. Manning. Deciding entailment and contradiction with stochastic and edit distance-based alignment. In *Proceedings of the First Text Analysis Conference (TAC 2008)*, 2009b.

Partha Pakray, Sivaji Bandyopadhyay, and Alexander Gelbukh. Lexical based two-way rte system at rte-5. In *Proceedings of the Text Analysis Conference (TAC 2009) Workshop*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Martha Palmer, Paul Kingsbury, and Daniel Gildea. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31(1):71–106, 2005.

Bo Pang, Kevin Knight, and Daniel Marcu. Syntax-based alignment of multiple translations: Extracting paraphrases and generating new sentences. In *HLT-NAACL*, pages 102–109, 2003.

Anselmo Peñas, Álvaro Rodrigo, and Felisa Verdejo. Overview of the answer validation exercise 2007. In *Proceedings of CLEF 2007 Working Notes*, Budapest, Hungary, 2007.

J. Platt. Machines using sequential minimal optimization. *Advances in Kernel Methods - Support Vector Learning*, 1998.

James Pustejovsky, Robert Gaizauskas, and Graham Katz. TimeML: Robust specification of event and temporal expressions in text. In *Fifth International Workshop on Computational Semantics*, 2003.

Chris Quirk, Chris Brockett, and William B. Dolan. Monolingual machine translation for paraphrase generation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 142–149, Barcelona, Spain, July 2004.

Chris Quirk, Raghavendra Udupa, and Arul Menezes. Generative models of noisy translations with applications to parallel fragment extraction. In *Proceedings of MT Summit XI*, Copenhagen, Denmark, 2007. URL `http://research.microsoft.com/nlp/publications/mtsummit2007_compcorp.pdf`.

Hans Reichenbach. *The Direction of Time*. Dover Publications, July 1999.

Yaroslav Riabinin. Recognizing textual entailment using logical inference: A survey of the pascal rte challenge. Online, 2008.

Á. Rodrigo, A. Peñas, and F. Verdejo. Overview of the answer validation exercise 2008. In *Working Notes of the CLEF 2008 Workshop*, Aarhus, Denmark, September 2008.

Álvaro Rodrigo, Anselmo Penãs, and Felisa Verdejo. Towards an entity-based recognition of textual entailment. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Dan Roth, Mark Sammons, and V.G.Vinod Vydiswaran. A framework for entailed relation recognition. In *Proceedings of ACL-IJCNLP 2009 Short Papers*, Singapore, Singapore, August 2009. Association for Computational Linguistics.

Mark Sammons, V.G.Vinod Vydiswaran, Tim Vieira, Nikhil Johri, Ming-Wei Chang, Dan Goldwasser, Vivek Srikumar, Gourab Kundu, Yuancheng Tu, Kevin Small, Joshua Rule, Quang Do, and Dan Roth. Relation alignment for textual entailment recognition. In *Proceedings of TAC 2009*, 2009.

Mark Sammons, Vinod Vydiswaran, and Dan Roth. Ask not what textual entailment can do for you... In *Proceedings of ACL 2010*, Uppsala, Sweden, 2010. Association for Computational Linguistics.

Frank Schilder and Christopher Habel. From temporal expressions to temporal information: Semantic tagging of news messages. *Proceedings of ACL'01 workshop on temporal and spatial information processing*, pages 65–72, 2001.

Satoshi Sekine. Automatic paraphrase discovery based on context and keywords between NE pairs. In *Proceedings of International Workshop on Paraphrase*, pages 80–87, Jeju Island, Korea, 2005.

Y. Shinyama and S. Sekine. Paraphrase acquisition for information extraction. In *Proceedings of International Workshop on Paraphrasing*, 2003.

Yusuke Shinyama, Satoshi Sekine, Kiyoshi Sudo, and Ralph Grishman. Automatic paraphrase acquisition from news articles. In *Proceedings of Human Language Technology Conference (HLT 2002)*, San Diego, USA, 2002. Association for Computational Linguistics.

Eyal Shnarch. Lexical entailment and its extraction from wikipedia. Master thesis, Computer Science Department, Bar-Ilan University, Israel, 2008.

Reda Siblini and Leila Kosseim. Using Ontology Alignment for the TAC RTE Challenge. In *Proceedings of the First Text Analysis Conference (TAC 2008)*, 2009.

Daniel Sleator and Davy Temperley. Parsing english with a link grammar. In *Proceedings of the Third International Workshop on Parsing Technologies*, 1993.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. Cheap and fast - but is it good? Evaluating non-expert annotations for natural language tasks. In *Proceedings of EMNLP*, 2008.

Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the 12th conference on computational natural language learning (CoNLL-2008)*, Manchester, UK, 2008.

Idan Szpektor and Ido Dagan. Learning entailment rules for unary templates. In *Proceedings of COLING 2008*, pages 849–856, Manchester, UK, 2008.

Idan Szpektor, Hristo Tanev, Ido Dagan, and Bonaventura Coppola. Scaling web-based acquisition of entailment relations. In *In Proceedings of EMNLP*, pages 41–48, 2004.

Idan Szpektor, Eyal Shnarch, and Ido Dagan. Instance-based evaluation of entailment rule acquisition. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 456–463, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P/P07/P07-1058`.

Idan Szpektor, Ido Dagan, Roy Bar-Haim, and Jacob Goldberger. Contextual preferences. In *Proceedings of ACL-08: HLT*, pages 683–691, Columbus, Ohio, June 2008. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P/P08/P08-1078`.

Marta Tatu, Brandon Iles, John Slavick, Adrian Novischi, and Dan Moldovan. COGEX at the second recognizing textual entailment challenge. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy, 2006.

L. Vanderwende, A. Menezes, and R. Snow. Microsoft research at rte-2: Syntactic contributions in the entailment task: an implementation. In *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, Venice, Italy, 2006.

Y. Versley, S. Ponzetto, Poesio M., V. Eidelman, A. Jern, J. Smith, X. Yang, and A. Moschitti. Bart: A modular toolkit for coreference resolution. In *Proceedings of ACL-08 Demonstration Session*, Columbus, Ohio, USA, 2008. Association for Computational Linguistics.

Stephan Vogel. Using noisy bilingual data for statistical machine translation. In *Proceedings of EACL*, 2003.

Ellen M. Voorhees. Contradictions and justifications: Extensions to the textual entailment task. In *Proceedings of ACL-08: HLT*, pages 63–71, Columbus, Ohio, USA, June 2008. Association for Computational Linguistics.

Rui Wang. Textual entailment recognition: A data-driven approach. Master's thesis, Saarland University, Saarbrücken, Germany, September 2007.

Rui Wang and Chris Callison-Burch. Cheap facts and counter-facts. In *Proceedings of NAACL-HLT 2010 Workshop on Amazon Mechanical Turk*, Los Angeles, California, 2010.

Rui Wang and Günter Neumann. Recognizing textual entailment using a subsequence kernel method. In *Proceedings of AAAI*, pages 937–942, 2007a.

Rui Wang and Günter Neumann. DFKI-LT at AVE 2007: Using recognizing textual entailment for answer validation. In *Proceedings of CLEF 2007 Working Notes*, Budapest, Hungary, September 2007b.

Rui Wang and Günter Neumann. Information synthesis for answer validation. In Carol Peters et al., editor, *CLEF 2008 Working Notes*, Aarhus, Denmark, 2008a. Springer Verlag.

Rui Wang and Günter Neumann. Relation validation via textual entailment. In Benjamin Adrian, Günter Neumann, Alexander Troussov, and Borislav Popov, editors, *1st International and KI-08 Workshop on Ontology-based Information Extraction Systems*, Kaiserslautern, Germany, 2008b. DFKI.

Rui Wang and Günter Neumann. An accuracy-oriented divide-and-conquer strategy for recognizing textual entailment. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, 2009. National Institute of Standards and Technology (NIST).

Rui Wang and Caroline Sporleder. Constructing a textual semantic relation corpus using a discourse treebank. In *Proceedings of the seventh international conference on Language Resources and Evaluation (LREC)*, Valletta, Malta, 2010.

Rui Wang and Yajing Zhang. Recognizing textual entailment with temporal expressions in natural language texts. In *Proceedings of the*

IEEE International Workshop on Semantic Computing and Applications (IWSCA-2008), pages 109–116, Incheon, Korea, Republic of, 2008. IEEE Computer Society.

Rui Wang and Yi Zhang. Recognizing textual relatedness with predicate-argument structures. In *Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*, Singapore, Singapore, 2009. Association for Computational Linguistics.

Rui Wang and Yi Zhang. MARS: A specialized rte system for parser evaluation. In *Proceedings of the SemEval-2010 Evaluation Exercises on Semantic Evaluation*, Uppsala. Sweden, July 2010.

Rui Wang and Yi Zhang. A multi-dimensional classification approach towards recognizing textual semantic relations. In *Proceedings of CI-CLing 2011*, Tokyo, Japan, February 2011.

Rui Wang, Yi Zhang, and Günter Neumann. A joint syntactic-semantic representation for recognizing textual relatedness. In *Text Analysis Conference TAC 2009 WORKSHOP Notebook Papers and Results*, pages 1–7, Gaithersburg, Maryland, USA, 2009. National Institute of Standards and Technology (NIST).

I. H. Witten and E. Weka Frank. Practical machine learning tools and techniques with java implementations. *Proceedings of the ICONIP/ANZIIS/ANNES*, 1999.

Dekai Wu and Pascale Fung. Inversion transduction grammar constraints for mining parallel sentences from quasi-comparable corpora. In *Proceedings of IJCNLP*, Jeju Island, Korea, 2005.

Mehmet Ali Yatbaz. Rte4: Normalized dependency tree alignment using unsupervised n-gram word similarity score. In *Proceedings of the Text Analysis Conference (TAC 2008) Workshop - RTE-4 Track*, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Deniz Yuret, Aydın Han, and Zehra Turgut. Semeval-2010 task 12: Parser evaluation using textual entailments. In *Proceedings of the SemEval-2010 Evaluation Exercises on Semantic Evaluation*, 2010.

Fabio Massimo Zanzotto, Marco Pennacchiotti, and Alessandro Moschitti. Pemoza submission to tac 2008. In *Proceedings of the Text*

Analysis Conference (TAC 2008) Workshop - RTE-4 Track, Gaithersburg, Maryland, USA, November 2009. National Institute of Standards and Technology.

Chen Zhang and Joyce Y. Chai. Towards conversation entailment: An empirical investigation towards conversation entailment: An empirical investigation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 756–766, MIT, Massachusetts, USA, October 2010. Association for Computational Linguistics.

Yi Zhang, Rui Wang, and Hans Uszkoreit. Hybrid Learning of Dependency Structures from Heterogeneous Linguistic Resources. In *Proceedings of the Twelfth Conference on Computational Natural Language Learning (CoNLL 2008)*, pages 198–202, Manchester, United Kingdom, 2008. Association for Computational Linguistics.

Shiqi Zhao, Haifeng Wang, Ting Liu, and Sheng Li. Pivot approach for extracting paraphrase patterns from bilingual corpora. In *Proceedings of ACL*, 2008.