PhD Dissertation

International Doctorate School in Information and Communication Technologies

DISI - University of Trento

Cross-Lingual Textual Entailment and Applications

Yashar Mehdad

Advisor:

Prof. Marcello Federico

Fondazione Bruno Kessler, Human Language Technology Research Unit.

March 2012

Abstract

Textual Entailment (TE) has been proposed as a generic framework for modeling language variability. The great potential of integrating (monolingual) TE recognition components into NLP architectures has been reported in several areas, such as question answering, information retrieval, information extraction and document summarization. Mainly due to the absence of cross-lingual TE (CLTE) recognition components, similar improvements have not yet been achieved in any corresponding cross-lingual application. In this thesis, we propose and investigate Cross-Lingual Textual Entailment (CLTE) as a semantic relation between two text portions in different languages. We present different practical solutions to approach this problem by i) bringing CLTE back to the monolingual scenario, translating the two texts into the same language; and ii) integrating machine translation and TE algorithms and techniques. We argue that CLTE can be a core technology for several cross-lingual NLP applications and tasks. Experiments on different datasets and two interesting cross-lingual NLP applications, namely content synchronization and machine translation evaluation, confirm the effectiveness of our approaches leading to successful results. As a complement to the research in the algorithmic side, we successfully explored the creation of cross-lingual textual entailment corpora by means of crowdsourcing, as a cheap and replicable data collection methodology that minimizes the manual work done by expert annotators.

Keywords

[Natural Language Processing, Textual Entailment and Paraphrasing, Cross-Lingual Textual Inference, Content Synchronization, Machine Translation, Crowd-Sourcing]

Acknowledgement

Few years back, while I was passionately tasting the sweet joy of teaching to young students in a college, I had to bear with the bitter taste of marking the projects and written exams. Being tired of such a boring end of semester moments, once that I was driving back home in a heavy traffic jam, the idea of automatically scoring the students papers and projects shined in my mind. Further crazy science fictional thoughts and day dreaming around this fancy technology led me to the Natural Language Processing field, and motivated me in driving along this road for 4-5 years, which resulted in this PhD thesis.

Though only my name appears on the cover of this dissertation, many people have contributed to and helped me in growing it up and its production. I owe my sincere gratitude to all those people who have made this thesis possible and because of whom such experience has been one that I will cherish forever.

I have been fortunate to have an advisor who gave me the freedom, courage and confidence to explore this path on my own, and at the same time the helpful guidance to recover when my steps failed. Marcello Federico taught me how to be precise, organized and focused. His support even during the last stages of my thesis and his management and coordination skills have been valuable lessons that I'm sincerely grateful for.

My deepest gratitude is to my true friend, colleague and co-advisor, Matteo Negri, who has been always there to listen, share ideas and give advices. Through long discussions, critical corrections, creative ideas on technical writings and presentations with him, I've learned priceless lessons during my PhD. I also owe him a debt of gratitude for carefully reading, commenting and countless revising my writings and slides in all stages of my work. Moreover, being always there, he made my stay in Trento and FBK one of the pleasant periods of my life.

I am grateful to Bernardo Magnini who first brought my attention to the topic of Textual Entailment, and facilitated my entry to this community and also to FBK. I am also thankful to him for giving me the opportunity of teaching, to add to the joy of living in an academic life and being closer to my goals.

Dr. Alessandro Moschitti is one of the most influential teachers that I had in my stay in Trento. He introduced me to Machine Learning and SVMs, specifically kernel methods. The works I've done in one of his courses resulted in a publication in a major conference. I am indebted to him for his continuous encouragement and guidance and for what I've learned from him during our

collaborations. I am also grateful to Dr. Fabio Massimo Zanzotto for our fruitful collaboration and the discussions that we had, sometimes for hours. I'd also like to express my deep gratitude to Dr. Farid Melgani, whose great help and guidelines resulted in my first ACL publication in a very early stage of my PhD.

I am also indebted to the members of the HLT group at FBK, who have provided a pleasant and stimulating environment to pursue my studies. Particularly, I would like to acknowledge the group of colleagues - Milen Kouylekov, Luisa Bentivogli, Claudio Giuliano, Christian Girardi, Mauro Cettolo, Elena Cabrio, Nicola Bertoldi, Alessandro Marchetti and Danilo Giampiccolo - for their precious collaboration.

I would like to acknowledge my PhD committee members, Dr. Ido Dagan, Dr. Miles Osborne, Dr. Lluis Marquez and Dr. Chistof Monz for taking their valuable time to travel to Trento and giving me the pleasure of having them in my final exam. I am also thankful to them for reading my dissertation, commenting on my views and helping me to improve and enrich my thesis.

Many friends have helped me to stay happy through these difficult years. I greatly value their friendship and I deeply appreciate their belief in me. I also like to thank Sofhia Rahim who helped me in proof-reading this manuscript.

Most importantly, none of this would have been possible without the love and patience of my family. My family to whom this dissertation is dedicated to, has been a constant source of love, concern, support and strength for all these years. I would like to express my heart-felt gratitude to my parents - Iran Mehdizadegan and Ali Mehdad - that have aided and encouraged me throughout my life. I also offer my deepest thanks and love to my brother and sister - Araz and Maral - whose love and support has sustained me during these years.

Contents

1 Introduction **1**

1.1 Context 1

1.2 Problem 3

1.3 Solution 4

1.4 Contributions 6

1.5 Structure of the Thesis 10

2 Recognizing Textual Entailment **11**

2.1 Textual Entailment 11

2.2 Datasets 14

2.3 Knowledge Resources 18

2.3.1 Lexical databases 18

2.3.2 Entailment and inference rules 20

2.3.3 Context sensitive lexical rules from wikipedia . . . 21

2.3.4 Paraphrase Tables as a Source of Knowledge 25

2.4 Approaches 27

2.4.1 Logic-based approaches to RTE 27

2.4.2 Transformation and Similarity based Approaches . 28

2.4.3 Supervised Learning Methods for RTE 30

2.5 Optimization 32

2.5.1 Optimizing TE Recognition Using PSO 32

2.5.2 Optimizing TE System Using Genetic Algorithm . 40

2.6 Syntactic Semantic Learning 47
2.6.1 Motivating Example 47
2.6.2 Lexical similarities 49
2.6.3 Integrating Semantic in Syntactic Tree Kernels . . . 51
2.6.4 Semantic Syntactic Tree Kernels for RTE 54
2.6.5 Experiments 56
2.6.6 Results 58
2.7 Summary 62

3 Cross-Lingual Textual Entailment 65
3.1 Introduction 65
3.2 CLTE 67
3.2.1 Definition 67
3.2.2 Approaches 68
3.3 Pivoting 72
3.3.1 Experiment 1: Feasibility Study 72
3.3.2 Experiment 2: Verification 75
3.4 Advanced 81
3.4.1 Exploiting Parallel Corpora for CLTE 83
3.4.2 Beyond Lexical Features 90
3.5 Summary 97

4 Content Synchronization 99
4.1 Introduction 99
4.2 CLTE 101
4.3 Experiments 105
4.3.1 Dataset 105
4.3.2 Features 107
4.3.3 Evaluation settings 108
4.3.4 Results 109

4.4 Open Issues . . . 113
4.5 Summary . . . 115

5 MT adequcy evaluation **117**
5.1 Introduction . . . 117
5.2 MT evaluation . . . 118
5.3 Predicting adequacy . . . 121
5.4 CLTE . . . 124
5.4.1 Features . . . 125
5.4.2 Dataset . . . 128
5.4.3 Algorithms and Approaches . . . 130
5.5 Results . . . 130
5.5.1 Adequacy and quality prediction . . . 130
5.5.2 Multi-class classification . . . 133
5.5.3 Recognizing “good” vs “bad” translations . . . 135
5.6 Summary . . . 138

6 CLTE dataset creation **141**
6.1 Introduction . . . 141
6.2 Crowdsourcing . . . 143
6.3 RTE3-derived CLTE dataset . . . 145
6.3.1 Methodology . . . 146
6.3.2 Experiments and lessons learned . . . 148
6.3.3 Results . . . 152
6.4 Content Synchronization . . . 153
6.4.1 Methodology . . . 155
6.4.2 Further Analysis . . . 161
6.5 Summary . . . 169

7 Conclusion **171**
7.1 Recapitulation 171
7.2 Future direction 173

Bibliography **177**

A List of Published Papers **201**

List of Tables

2.1 RTE datasets. 17
2.2 Comparing accuracy results over the RTE-5 dataset. . . . 24
2.3 Coverage of rule repositories over the RTE-5 dataset. . . . 25
2.4 Accuracy results on RTE using different lexical resources. . 26
2.5 Optimized and unoptimized cost schemes comparison. . . . 40
2.6 RTE results (acc. for RTE1-RTE5, F-meas. for RTE6). . . 46
2.7 Accuracy comparison using different semantic rules. 58
2.8 Lexico-syntactic kernels comparison. 60
2.9 Coverage of the different resources. 61
2.10 Efficiency comparison. 61
2.11 Comparison with other approaches to RTE 62

3.1 CLTE feasibility study. 74
3.2 Pivoting approach using different lexical resources. 80
3.3 CLTE using different bilingual lexical resources. 89
3.4 Dependency Relation (DR) matching. 91
3.5 Semantic Phrase Table (SPT) matching. 93
3.6 CLTE accuracy results over the RTE3 derived dataset. . . 96

4.1 CLTE for content synchronization accuracy results. 111

5.1 *Adequacy-oriented* features v.s. the *quality* features. 124
5.2 Pearson's correlation over the 16K dataset. 131
5.3 Pearson's correlation over the WMT07 dataset. 133

5.4 Multi-class classification over 16K dataset. 134
5.5 Multi-class classification over WMT07 dataset. 135
5.6 Binary classification over 16K dataset. 136
5.7 Binary classification over WMT07 dataset. 137

6.1 Creating a RTE3-derived CLTE dataset. 151
6.2 The monolingual dataset creation pipeline. 162

List of Figures

1.1 Contributions' timeline. 7

2.1 EDITS-GA framework. 43
2.2 A syntactic parse tree along with some of its fragments. . . 52

3.1 CLTE approaches 69
3.2 N-best Moses translations accuracy. 75
3.3 Using a phrase table for CLTE. 84
3.4 Combining phrase and paraphrase tables for CLTE. 85
3.5 Phrase tables with different pruning thresholds. 88

4.1 Automatic content synchronization framework. 102

6.1 Content Synchronization corpus creation process. 157
6.2 Sentence modification and TE annotation pipeline. 159

Chapter 1

Introduction

1.1 Context: Textual Entailment and Inference

Natural languages allow to express the same meaning in many possible ways, making automatic understanding particularly challenging. Almost all computational linguistics tasks such as information retrieval (IR), question answering (QA), information extraction (IE), text summarization and machine translation (MT) have to cope with this phenomenon. textual entailment recognition was proposed by Dagan & Glickman [2004] as a generic NLP task in order to overcome the problem of lexical, syntactic and semantic variability in natural languages. In 2005, The recognizing textual entailment (RTE) Challenge has been launched by Dagan et al. [2005], defining textual entailment (TE) as a task for automatic systems. Given a text T and a hypothesis H, the task consists of deciding if the meaning of H can be inferred from the meaning of T. The following examples show T-H pairs for which the entailment relation holds (Example 1) or not (Example 2):

Example 1.

T*: Euro-Scandinavian media cheer Denmark vs Sweden draw.*

H*: Denmark and Sweden tie.*

Entailment*: YES*

Example 2.

***T**: Oracle had fought to keep the forms from being released.*

***H**: Oracle released a confidential document.*

***Entailment**: NO*

In the many evaluation campaigns that in recent years addressed the TE recognition problem, complex definitions of the task have been proposed. The released datasets reflect the long-term objective of creating more natural evaluation settings. These include the formulation of TE as a search task[1] (*i.e.* finding all the sentences in a set of documents that entail a given hypothesis), the use of TE to approach the Answer Validation Exercise[2] (emulate human assessment of QA responses and decide whether an answer to a question is correct or not according to a given text), and the very recent effort to explore multi-directional TE recognition[3] (moving from YES/NO to directional entailment judgements such as Forward, Backward and Bidirectional). Consequently, a large number of methods and resources for TE has been published or released.

Even though the research community is currently considering a number of NLP applications under multi-lingual or cross-lingual perspectives (including QA, IR, IE, and text summarization), not much is being done in the area of TE recognition. The first concrete attempt to move from the traditional English evaluation datasets is represented by the 2009 edition of the EVALITA Challenge,[4] which hosted a TE recognition task for Italian. However, cross-language TE recognition capabilities have been completely disregarded, until the seminal work presented in this thesis.

[1] RTE: http://www.nist.gov/tac/2010/RTE/

[2] AVE: http://nlp.uned.es/clef-qa/ave/

[3] NTCIR-9 RITE: http://artigas.lti.cs.cmu.edu/rite/

[4] http://evalita.fbk.eu/index.html

1.2 The Problem: Cross-Lingual Textual Entailment

The explosion of multilingual content in the web provides users with the opportunity to access and publish information about a given topic in their own language. The dramatic growth of content published in languages other than English demonstrates the high demand of multilingual and cross-lingual NLP applications. The growth rate of Chinese, Spanish and Portuguese as languages used in the web (1,478.7%, 807.4% and 990.1% respectively, between 2000-2011)[5] confirms the need of cross-lingual technology to help users bridge the language barrier to access information and communicate with each other over the internet.

The great potential in taking advantage of monolingual TE recognition components into NLP applications has been reported in several research works (*e.g.* [Roth et al., 2009; Mirkin et al., 2009b; Zhang & Chai, 2010]). However, mainly due to the absence of cross-lingual TE recognition components, similar improvements have not been achieved yet in any cross-lingual application. As a matter of fact, despite the great deal of attention that TE has received in recent years and the emerging research in multilingual scenarios, interest for cross-lingual extensions has not been in the mainstream of TE research.

Building on these considerations, this thesis aims at proposing and exploring for the first time cross-lingual textual entailment (CLTE) as a way to perform semantic inference across languages. The CLTE task is inherently difficult, as it adds multilinguality issues to the complexity of semantic inference at the textual level. For instance, the reliance of current monolingual TE systems on lexical resources (*e.g.* WordNet, VerbOcean, FrameNet) and deep processing components (*e.g.* syntactic and semantic parsers, co-reference resolution tools, temporal expressions recognizers and

[5]Reported from http://www.internetworldstats.com/stats7.htm

normalizers) has to confront, at the cross-lingual level, with: *i)* the limited availability of lexical/semantic resources covering multiple languages, *ii)* the limited coverage of the existing ones, and *iii)* the burden of integrating language-specific components into the same cross-lingual architecture. Despite the multilingual challenges posed by this task, research can now benefit from recent advances in other fields, especially machine translation, and the availability of large amounts of parallel and comparable corpora in many languages. All these resources can potentially help in developing inference mechanisms for multilingual data.

From the theoretical point of view, this thesis aims at building on the integration of semantics and MT resources and technology to tackle the difficulties of the CLTE task.

From the application point of view, this thesis aims at exploring the potential of CLTE in two different scenarios: *i)* the automatic synchronization of the topically related content text portions in tidy multilingual environments (such as wikis); and *ii)* the automatic estimation of the adequacy of MT systems' output without using reference translations.

1.3 The Proposed Solutions

This thesis describes two main methodologies to approach CLTE:

1. A "*basic approach*", that brings CLTE back to a monolingual task by translating H into the language of T, or vice-versa.

2. An "*advanced approach*", that embeds cross-lingual processing techniques inside the CLTE recognition process.

Building on our experience in monolingual English RTE approaches, initially we explored the simplest approach to CLTE. Such approach consists in adding a MT component to the front-end of an existing TE engine.

For instance, let the hypothesis H be translated into the language of T, and then run the TE engine on the T and the translation of the H. There are several good reasons to follow this divide-and-conquer approach, apart from some drawbacks. Decoupling the cross-lingual and the entailment components results in a simple and modular architecture that, according to well known software engineering principles, is easier to develop, debug, and maintain. Moreover, a decoupled CLTE architecture would allow for easy extensions to other languages, as it just requires extra MT systems. Along with the same idea of pivoting through English, in fact, the same TE system can be employed to perform CLTE between any language pair, once MT is available from each language into English. Despite the advantages in terms of modularity and portability of the architecture and the promising experimental results achieved in our early works, the "*basic approach*" suffers from being dependent on the availability of MT components and to the quality of the translations. As a consequence, on one side, translation errors propagate into the TE engine thus hampering the entailment decision process. On the other side, such unpredictable errors reduce the possibility to control the behaviour of the engine, and devise ad-hoc solutions to specific entailment problems.

The idea behind the "*advanced approach*" to CLTE is to move towards a cross-lingual TE approach that takes advantage of a tighter integration of MT and TE algorithms and techniques. This could result in methods for recognizing TE across languages avoiding dependencies on external MT components; thus, eventually gaining full control of the system's behaviour. Along with this direction, we started from the acquisition and use of lexical knowledge, which represents the basic building block of any TE system. As the next step, we integrated linguistically motivated features (syntactic and semantic) to improve the state-of-the-art in the lexical CLTE approach.

The adoption of our CLTE approaches in different application scenarios (content synchronization and MT evaluation), aims at proving their effectiveness to real-world problems.

1.4 Innovative Aspects and Contributions

The work described in this thesis covers different topics related to TE research, ranging from contributions to monolingual TE in terms of algorithms and resources, to the proposal and exploitation of CLTE as a new task, and the design of novel data acquisition methods. Figure 1.1 shows a Gantt chart, which demonstrates the problems addressed and the main contributions over the completion time of this thesis. Such contributions can be summarized as follows:

Monolingual TE: methods to optimize the distance-based TE approaches and systems. In [Mehdad, 2009; Mehdad & Magnini, 2009a] we proposed a stochastic method based on Particle Swarm Optimization (PSO), to estimate the cost of edit distance operations for textual entailment problem. By means of PSO, we tried to learn the optimal cost for each edit distance operation in order to improve the prior textual entailment models. Besides the automatic learning of operational costs, another added advantage of such method is that it presents the ability to investigate the cost values to better understand how to approach TE with edit distance algorithms. Along with the same direction, in [Kouylekov et al., 2011], we used Genetic Algorithms to automatically obtain the most promising configuration for the EDITS RTE system [Kouylekov & Negri, 2010], avoiding the exhaustive exploration and testing all possible configurations.

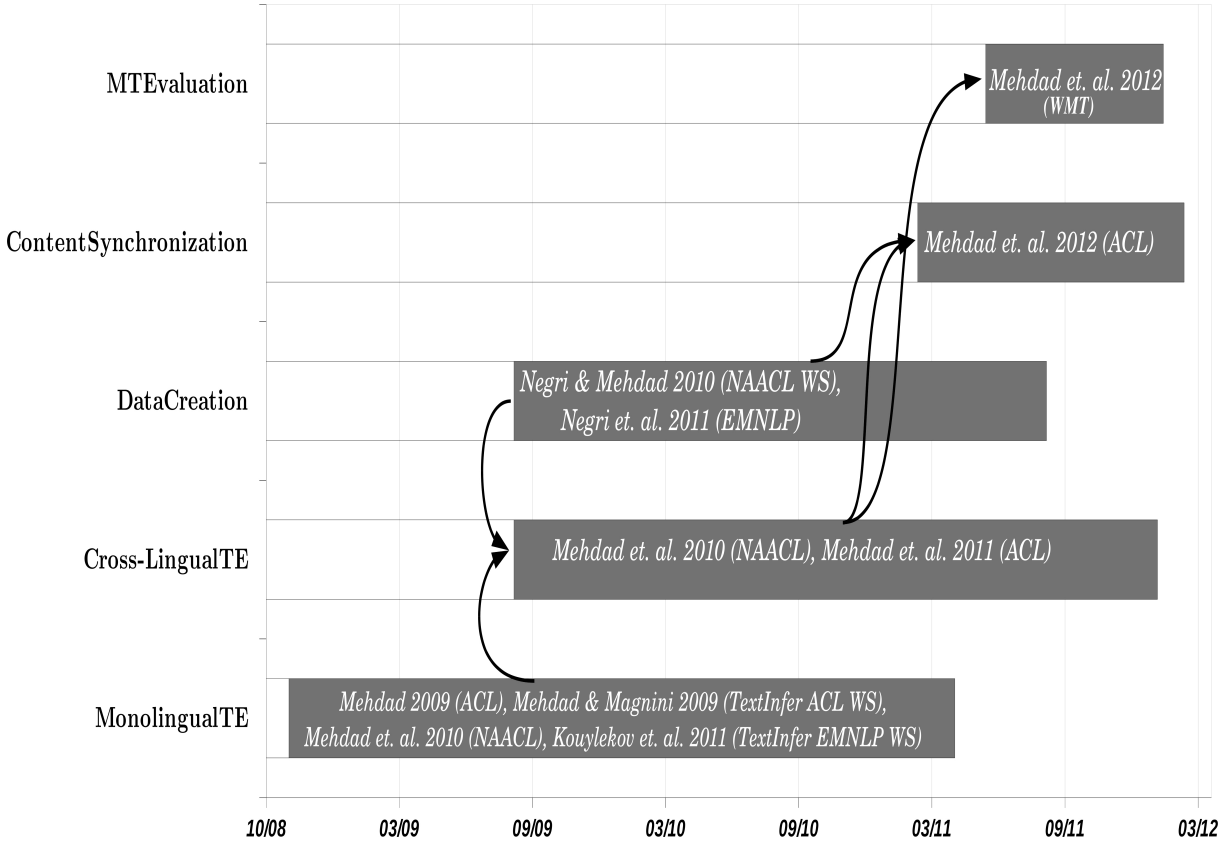

Figure 1.1: Achievements of this thesis in terms of major publications over the completion time. This chart shows the start and completion dates and dependencies of the chapters that are detailed in this thesis.

Monolingual TE: extraction of context-sensitive entailment rules from Wikipedia. In [Kouylekov et al., 2010a; Mehdad et al., 2010a] we proposed a method to embed context sensitivity into lexical entailment rules. Such method is based on computing similarity scores between words/phrases over Wikipedia by means of Latent Semantic Analysis (LSA). Our results demonstrate that applying entailment rules extracted from Wikipedia, we gain a higher coverage as well as a better performance in our entailment framework.

Monolingual TE: syntactic/semantic learning for textual en-

tailment recognition. In [Mehdad et al., 2010a] we propose models to effectively use syntactic and semantic information in RTE, without requiring either large automatic rule acquisition or hand-coding. These models exploit lexical similarities to generalize lexical-syntactic rules automatically derived by supervised learning methods. In more detail, syntax is encoded in the form of dependency parse trees whereas similarities are defined by means of WordNet similarity measures or Latent Semantic Analysis (LSA) applied to Wikipedia or to the British National Corpus (BNC). The joint syntactic/semantic model is realized by means of novel tree kernels, which can match subtrees whose leaves are lexically similar or related.

CLTE: proposal of a new research problem. In [Mehdad et al., 2010b] we proposed and investigated for the first time, a *cross-lingual* extension of TE where we assume that T and H are written in different languages, to allow for semantic inference across languages. Besides a feasibility study, we also presented two possible approaches to this task, evaluating advantages and disadvantages of each solution.

CLTE: use of parallel corpora. In [Mehdad et al., 2011] we explored the use of bilingual parallel corpora as a source of lexical knowledge for cross-lingual textual entailment. Our hypothesis is that, in spite of the inherent difficulties of the task, phrase and paraphrase tables extracted from parallel data allow to capture both lexical relations between single words, and contextual information useful for inference. Our experiments prove the potential of parallel corpora to approach cross-lingual textual entailment.

Applications: automatic content synchronization. As the first application framework, we addressed the task of synchronizing the content

of two documents about the same topic [Mehdad et al., 2012a; Mehdad & Negri, 2012], written in different languages, by adopting a solution based on CLTE. As a subtask of this problem, the identification of semantically equivalent text portions and more informative fragments in one of the two pages has been cast as an application-oriented variant of the CLTE task where entailment relations have to be checked in all possible directions (*i.e.* from text to hypothesis and vice-versa). Experimental results (under peer-reviewing process) demonstrate the benefits of adopting CLTE to approach such application scenario.

Applications: automatic adequacy evaluation of MT output. As the second application framework, in [Mehdad et al., 2012b] we proposed a methodology based on CLTE to estimate the adequacy of MT output without using reference translations. By casting the problem as a cross-lingual textual entailment recognition task, we could: *i)* avoid using costly hand-crafted reference translations, and *ii)* integrate semantics into MT evaluation in order to complement the shallow methods currently used, and overcome their limitations. The positive results of our work (under peer-reviewing process) show the effectiveness of CLTE components in dealing with such application scenario.

Data acquisition methods: crowd-sourcing the creation of CLTE corpora. In [Negri et al., 2011; Negri & Mehdad, 2010] we devised cost-effective methodologies to create cross-lingual textual entailment corpora, based on crowd-sourcing. Our results and released CLTE datasets confirmed that adopting these methodologies we can address the issues related to the shortage of data and the high costs for their creation in the CLTE scenario.

1.5 Structure of the Thesis

The thesis is structured as follows:

Chapter 2 gives an overview of textual entailment problem presenting : *i)* the state-of-the-art in TE research, *ii)* the Recognizing Textual Entailment (RTE) challenge, *iii)* possible TE applications, *iv)* the lexical and semantic resources used for RTE, and *v)* our novel contributions to monolingual TE.

Chapter 3, the core of our work, introduces the Cross-Lingual TE (CLTE) problem followed by a feasibility study, presenting the possible solutions we advocate in terms of theoretical insights and algorithms. The experimental setups, datasets and results are reported afterwards.

Chapter 4 presents content synchronization as a possible interesting application of CLTE. This chapter describes the framework we designed to tackle the problem, the experiments and results achieved over different datasets we created.

Chapter 5 presents another interesting application for CLTE: the automatic evaluation of machine translation adequacy without reference translations. We report extensive experiments on two different datasets and promising results achieved in several experimental settings.

Chapter 6 describes cost-effective and replicable methodologies to create and annotate CLTE and content synchronization datasets, taking advantage of crowdsourcing. Such methodologies have been successfully exploited to build the CLTE and content synchronization datasets used in our experiments.

Chapter 7 draws the conclusions and suggests possible future works.

Chapter 2

Recognizing Textual Entailment

2.1 Textual Entailment

Dagan & Glickman [2004] proposed the notion of Textual Entailment (TE) as a generic framework for modeling language variability and capturing major semantic inference needs across applications in NLP. TE is defined as a relationship between a coherent textual fragment T and a language expression or hypothesis H. Entailment holds, *i.e.* T $\Rightarrow$ H, if the meaning of H can be inferred from the meaning of T. This relationship is directional and asymmetric, since the meaning of one expression may usually entail the other, but not vice versa, unlike the semantic equivalence relation which is symmetric.

In 2005, the Recognizing Textual Entailment (RTE) Challenge was launched by Dagan et al. [2005], defining Textual Entailment as a task for automatic systems. Given two texts T and H, the task consists in deciding if the meaning of H can be inferred from the meaning of T. Example 1 shows a T-H pair for which the entailment relation holds:[1]

Example 1.

***T**: In the end, defeated, Antony committed suicide and so did Cleopatra,*

[1]This example is extracted from the official RTE dataset.

according to legend, by putting an asp to her breast.
***H**: Cleopatra committed suicide.*

At present, textual entailment is considered an interesting and challenging topic within the NLP community, due to its many potential applications. The PASCAL Network of Excellence[2] promoted a generic evaluation framework covering semantic-oriented inferences for several NLP applications, which led to launch the Recognizing Textual Entailment (RTE) Challenge.[3]

Many research areas such as information retrieval, question answering, information extraction, text summarization and machine translation have to cope with different kinds of inference mechanisms, closely related to the entailment notion. In this direction, some works tried to address different NLP tasks with textual entailment in order to benefit from a semantic inference framework, and to potentially improve their performances [Glickman, 2006].

In Question Answering (QA) some reasoning is needed to identify which texts are potentially informative answers for a given question. For example, given the question type *"What is the height of X?"* textual entailment can be performed to infer that texts such as *"X is N meters tall"* are informative to this question, while texts such as *"X is N kilograms"* are not. On the other hand, given the question *"Where is Eiffel tower?"*, it would be very helpful to discriminate that the answers should carry information related to *Paris* or *France*, and not to the hotel named the same, but located in Las Vegas.

The following examples try to better clarify the role of TE in QA applications. Harabagiu & Hickl [2006] applied textual entailment to either

[2] http://www.pascal-network.org/
[3] http://pascallin.ecs.soton.ac.uk/Challenges/RTE/

filter or rank answers returned by a QA system and reported about 20% improvement in performance. Bogdan et al. [2008] implemented a TE based approach to QA and proved that the system can be used in both monolingual and multilingual settings. Finally, Wang & Neumann [2008b] took advantage of TE in their answer validation framework and reported promising results.

Information Extraction (IE) is another NLP task which also recently benefited from the application of TE methods. Among them, Wang & Neumann [2008c] reported the feasibility of using TE for the relation validation task. Moreover, Roth et al. [2009] argued that TE is necessary to approach relation recognition task by proposing a scalable TE architecture.

In Information Retrieval (IR), a typical problem is the lexical gap between a query and a document. Van Rijsbergen [1979] proposed a method to fill this gap by taking advantage of TE, to infer the query from the document. Additionally, Kotb [2006] proposed an approach to retrieve not only textual documents that have the queried keywords, but also to discover semantically equivalent or entailed documents from the given keywords.

Concerning text summarization, Dragomir [2000] reported that entailment is among the cross-document relations that can hold between segments in a document. Once detected, this would provide a means to reduce redundancy in summarization. Over and above that, Doina et al. [2008] proved that utilizing TE for segmentation before summarization could improve the quality of final summaries. Moreover, one of the goals of the proposed *search task* in RTE is to analyze the potential impact of textual entailment on the summarization task, as proposed by the summarization community in the 2008 Text Analysis Conference (TAC).[4]

On top of that, TE has been proposed as an effective method for automatic evaluation of Machine Translation (MT). [Padó et al., 2008] used

[4]http://www.nist.gov/tac/2009/Summarization/index.html

this technique and proved that entailment-based MT evaluation metrics can keep up with the constantly improving quality of MT output, which is difficult to measure with surface-oriented methods. Furthermore, Mirkin et al. [2009b] proposed a new entailment-based approach for addressing the translation of unknown terms in MT. By applying this approach with lexical entailment rules extracted from WordNet, they improved the quality of translations produced by their MT system.

Among recent novel applications of TE, Agerri [2008] proposed to broaden the coverage of TE systems for the benefit of research on metaphors. Zhang & Chai [2010] investigated the problem of conversation entailment, that is automatically inferring the hypotheses from conversation scripts by examining two levels of representations of conversation utterances: syntactic and semantically augmented structures. In addition, as an interesting application, Bos & Oka [2007] focused on linguistic and inferential aspects of the human-robot communication via TE.

2.2 Datasets

In the previous section we defined the notion of TE [Dagan & Glickman, 2004], and overviewed the problems of natural language processing and understanding that can benefit from this framework. In this section we focus on the datasets and evaluation framework development for the RTE challenge.

Understanding the strong need of setting a benchmark for the development and evaluation of methods that address the TE problem, the PASCAL Network of Excellence started to organize an evaluation framework, casting the Recognizing Textual Entailment (RTE) Challenge in 2005 [Dagan et al., 2005]. The goal of this evaluation campaign is to promote the development of entailment recognition systems to provide generic modules

across applications.

Since 2005, this initiative has been repeated every year: RTE-1 [Dagan et al., 2005], RTE-2 [Roy Bar-Haim et al., 2006], RTE-3 [Giampiccolo et al., 2007], RTE-4 [Giampiccolo et al., 2008], RTE-5 [Bentivogli et al., 2009], RTE-6 [Bentivogli et al., 2010b] and RTE-7 [Bentivogli et al., 2011]. Since 2008, RTE has been proposed as a track at the Text Analysis Conference (TAC),[5] jointly organized by the National Institute of Standards and Technology[6] and CELCT.[7]

Each year, the organizers create development and test datasets, containing pairs of text fragments (the text T and the hypothesis H), with their relative entailment judgments. In this framework systems should determine whether the meaning of H is entailed, *i.e.* can be inferred from T (*e.g.* Example 2 from RTE-4). Since RTE-3 till RTE-5, systems could optionally make a further distinction between no entailment pairs: *i)* the entailment does not hold because the content of H is contradicted by the content of T (CONTRADICTION, *e.g.* Example 3 from RTE-3), and *ii)* the entailment cannot be determined because the truth of H could not be verified on the basis of the content of T (UNKNOWN, *e.g.* Example 4 from RTE-3). The distribution according to the three-way way annotation has been fixed to: 50% entailment, 35% unknown, and 15% contradiction pairs.

Example 2

***T**: A judge in Texas has signed an order allowing parents to take home more than 400 children who had been removed from a polygamist sect. Parents were set to begin collecting their children, who were seized from the sect's ranch by state authorities in April.*

***H**: US sect children are sent home.*

[5] http://www.nist.gov/tac/about/index.html
[6] http://www.nist.gov/index.html
[7] http://www.celct.it/

***Entailment**: YES*

Example 3

***T**: Stolen Warhol works recovered: Amsterdam police said Wednesday that they have recovered stolen lithographs by the late U.S. pop artist Andy Warhol worth more than $1 million. Dali's paintings are still missing.*
***H**: Millions of dollars of art were recovered, including works by Dali.*
***Entailment**: CONTRADICTION*

Example 4

***T**: Alex Dyer, spokesman for the group, stated that Santarchy in Auckland is part of a worldwide phenomenon.*
***H**: Alex Dyer represents Santarchy.*
***Entailment**: UNKNOWN*

Table 2.1 shows the exact number of pairs in each RTE edition dataset. It's worth mentioning that in RTE-6 and RTE-7, the traditional main task was replaced by the task of recognizing textual entailment within a corpus, situated in the text summarization setting, to challenge the systems by proposing a dataset which reflects the natural distribution of entailment in a corpus [Bentivogli et al., 2010b]. In such task, given a corpus, a hypothesis H, and a set of "candidate" sentences retrieved by Lucene[8] from that corpus (Ts), RTE systems are required to identify all sentences (Ts) that entail H [Bentivogli et al., 2011] (*e.g.* Example 5 from RTE-7).

Example 5

[8] Apache Lucene is a high-performance, full-featured text search engine library written entirely in Java: urlhttp://lucene.apache.org/core/

RTE-1		RTE-2		RTE-3		RTE-4		RTE-5		RTE-6		RTE-7	
Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
567	800	800	800	800	800	-	1000	600	600	15,955	19,972	21,420	22,426

Table 2.1: Number of main task pairs in each edition of RTE evaluation campaign.

***T**: French sports daily L'Equipe reported Tuesday that Lance Armstrong used the performance-enhancing drug EPO to help win his first Tour de France in 1999, a report the seven-time Tour winner vehemently denied.*
***H**: Lance Armstrong is a Tour de France winner.*

Moreover, starting from the intuition that detecting entailment relations by relying on linguistic foundations should make the systems stronger, Bentivogli et al. [2010a] proposed a methodology for the creation of specialized TE datasets. Such approach is made of monothematic T-H pairs in which a certain phenomenon underlying the entailment relation is highlighted and isolated. They also provided the annotation of RTE-5 data with the linguistic phenomena.

In addition, in the same context, Sammons et al. [2010] proposed a linguistically-motivated analysis of entailment data based on a step-wise procedure to resolve entailment decisions. The authors carried out a feasibility study applying the procedure to 210 examples from the RTE-5 collection, marking for each example the entailment phenomena that are required for the inference.

Last but not least, as one of the novel contributions of this PhD thesis, in Chapter 6 we address the creation of textual entailment corpora by means of crowd-sourcing aiming at defining a cheap and replicable data collection methodology that minimizes the manual work done by expert annotators.

2.3 Knowledge Resources

An important aspect in dealing with the Textual Entailment problem is represented by the amount of knowledge required to correctly handle the input T-H pairs. To address this issue, the main sources of knowledge that have been used for RTE are categorized into: *i) lexical databases*, and *ii) entailment and inference rules*.

The reported usage of entailment and inference rules is considerably lower than the wide usage of lexical resources. One of the main reasons is the limitation of entailment rules both in terms of availability and coverage. In addition, exploiting such resources efficiently needs more investigation and advanced algorithms and approaches.

In order to evaluate the contribution of different resources to the systems' performances, ablation tests were introduced for RTE-5, RTE-6 and RTE-7 main tasks. Ablation tests consist in removing one module at a time from a system, and re-running the system on the test set. Unluckily, as emerges from the ablation tests reported in [Bentivogli et al., 2009, 2010b, 2011], even the most common resources proved to have a positive impact on some systems and a negative impact on others. However, WordNet is the most commonly used lexical resource for TE.

2.3.1 Lexical databases

There are four categories of lexical knowledge resource which has been used in RTE:

- WordNet [Fellbaum, 1998] is employed in most of the RTE systems in different ways mainly to compute a similarity score between two words using the semantic links, *e.g.*, synonyms, hyponym, hypernyms and etc. (*e.g.* [Galanis & Malakasiotis, 2009; Clark & Harrison, 2009]).

Extensions of Wordnet such as EuroWordNet[9], and eXtended WordNet[10] have been also exploited by Bos [2005]; Tatu & Moldovan [2007].

- VerbNet [Schuler, 2005], and VerbOcean [Chklovski & Pantel, 2004] are mostly used in order to obtain relations between verbs (*e.g.* in [Balahur et al., 2008; Mehdad et al., 2009b; Ferrández et al., 2009]). In particular, two verbs are related if they belong to the same VerbNet class or subclass; or if they satisfy one of the VerbOcean relations: *similarity*, *strength*, or *happens-before*.

- FrameNet [Baker et al., 1998] was integrated in some systems (*e.g.* [Delmonte et al., 2007]) in different ways. Ferrández et al. [2009] defined a similarity score based on FrameNet, while Tatu & Moldovan [2005] derived new semantic information by using FrameNet's frame elements identified in text. However, most of the works were limited in using FrameNet probably because of its restricted coverage or the difficulties in modeling its information (see [Burchardt et al., 2009]).

- Wikipedia, as a large web corpus, has been used by many systems, to extract lexical knowledge or entailment rules. Shnarch [2008] used Wikipedia to create an extensive resource of 8 million lexical entailment rules which has been exploited in [Bar-Haim et al., 2008]. Moreover, Wikipedia has been used by [Li et al., 2009a] mainly for named-entity resolution, since there are different references to the same entity with a high coverage. Finally, Mehdad et al. [2010a]; Kouylekov et al. [2010a] used lexical rules extracted from Wikipedia to measure lexical similarity. This work is discussed in Section 2.3.3 in more details as one of the contributions of this PhD thesis.

[9] http://www.illc.uva.nl/EuroWordNet/
[10] http://xwn.hlt.utdallas.edu/

2.3.2 Entailment and inference rules

Textual entailment and inference rules are usually automatically acquired rewriting rules. In fact, due to the coverage problem, hand crafted rules are in general not sufficient for a TE system. The most widely used repository of entailment rules is DIRT [Lin & Pantel, 2001], containing a set of inference rules, represented as pairs of directional relations between two text patterns with variables (*e.g.* *"X put emphasis on Y"* ⇒ *"X pay attention to Y"*).

Beyond DIRT, Szpektor & Dagan [2008] investigated two approaches for unsupervised learning of unary rule (*i.e.* one-directional entailment rules) extraction and compared these methods with a learning method to extract binary rules (i.e. bidirectional entailment rules). Their results show that the learned unary rules outperform the binary rules. Aharon et al. [2010] proposed an algorithm that generates inference rules between predicates from FrameNet and proved to be more efficient than WordNet. Furthermore, [Berant et al., 2011] implemented an algorithm that utilizes transitivity constraints to learn a globally-optimal set of entailment rules for typed predicates by modeling the task as a graph learning problem.

Although some systems in the RTE challenges used DIRT (*e.g.* [Mirkin et al., 2009a; Bos & Markert, 2005]) or other mentioned entailment rules as a source of knowledge, the experimental results did not report any significant contribution. This might be due to the noise introduced in automatically-acquired rules or the challenge of rule application.

In addition, as one of the contributions of this PhD thesis, in [Mehdad et al., 2011] we show that using parallel corpora to extract paraphrase rules can help improving the results achieved with other sources of lexical knowledge in RTE task. This work is explained in Section 2.3.4.

2.3.3 Context sensitive lexical rules from wikipedia

Wikipedia, as a source of lexical entailment rules, offers at least two advantages over other resources. The first is coverage: with more than 3.000.000 articles, Wikipedia covers the vast majority of concepts potentially appearing in any RTE dataset. This is particularly evident with named entities (*e.g.* instances of the categories PERSON or LOCATION), whose coverage in Wikipedia is much larger than in any other available source of lexical knowledge. The second advantage is context sensitivity: Wikipedia allows to consider the context (*i.e.* the actual content of the articles) in which rule elements tend to appear.

To embed context in our rules, we train a Latent Semantic Analysis (LSA) model over Wikipedia and use it to score all possible word pairs that appear in the T-H pairs of an RTE dataset. To this aim we use the jLSI (java Latent Semantic Indexing) tool[11] to measure the relatedness between all the terms in a T-H pair. We created the model from the 200,000 most visited Wikipedia articles, after cleaning unnecessary markup tags. Cleaned articles are used as documents for creating the term-by-document matrix. Then, we empirically estimate over the training data a relatedness threshold in order to filter out all the pairs of terms featuring low similarity, thus obtaining a set of pairs where the first term entails the second one with a high probability.

The threshold was empirically estimated running a set of experiments to select the subset of rules that best performs on training data. This could result in a good trade-off between precision and coverage of the extracted rules. Though higher thresholds could increase precision, leading to more accurate rules, the reduced amount of extracted rules would directly affect coverage, causing an overall performance decrease.

[11] Available at http://tcc.itc.it/research/textec/tools-resources/jLSI.html

In order to compare rule repositories obtained from different resources in the RTE task and validating the usefulness of the rules extracted from Wikipedia, we used EDITS (Edit Distance Textual Entailment Suite [Kouylekov & Negri, 2010]), a freely available open source tool for recognizing textual entailment developed by FBK-irst, and our novel syntactic/semantic tree kernel system developed by Mehdad et al. [2010a]. The mentioned systems will be described in Section 2.4.

Since our objective is to compare the utility of the lexical knowledge extracted from Wikipedia with other resources, each experiment has been carried out with the best configuration of EDITS in RTE-5 (the one used for RTE-5 submission, which is thoroughly described in [Mehdad et al., 2009b]) and different configurations of tree-kernel based system used in RTE-5, which is thoroughly described in [Mehdad et al., 2009a]. In this section, we only describe the results and experiments with EDITS, while the interesting findings with a syntactic/semantic tree kernel system are presented in Section 2.6.

In our experiments, we compared the performance achieved over the RTE-5 dataset by exploiting the following lexical rule repositories:

WIKI: Out of the original 199,217 rules extracted from Wikipedia, we estimated a threshold over training data to filter out rules with lower reliability. As a result, 58,278 rules have been retained.

WN: 1,106 rules have been extracted from WordNet for each pair of terms (w_1 in T and w_2 in H) that are connected by the synonym or hypernym relations. More specifically, given a word w_1 in T, a new rule $[w_1 \Rightarrow w_2]$ is created for each word w_2 in H that is a synonym or an hypernym of w_1.

VO: 192 rules have been extracted from VerbOcean. Rules are collected for each pair of verbs (v_1 in T and v_2 in H) that are connected by the [*stronger-than*] relation. More specifically, given a verb v_1 in T, a new rule $[v_1 \Rightarrow v_2]$ is created for each verb v_2 in H that is connected to v_1 by the [*stronger-than*] relation (*i.e.* when [v_1 *stronger-than* v_2]). Though potentially useful, transitive closure is not considered due to the high level of noise introduced by verb ambiguities.

DEP: rules are collected from the thesauri of dependency based similarities described in [Lin, 1998], and available at Dekang Lin's website[12]. More specifically, given a word w_1 in T, a new rule $[w_1 \Rightarrow w_2]$ is created for each word w_2 in H that is related to w_1 in the thesauri. Out of the 5,432 rules extracted from Lin's dependency thesaurus, we estimated a threshold to filter out those with lower reliability.

PROX: in the same way, out of 8,029 original rules extracted from the Lins proximity thesaurus, only 236 have been retained after filtering.

Table 2.2 reports the accuracy results we achieved over RTE-5 data (both on the development and test sets), showing that Wikipedia rules outperform all the other rule sets, with accuracy improvements over the test set ranging from 2.5% to 5.2%.

These results demonstrate that applying entailment rules extracted from Wikipedia, we gain a higher coverage as well as a better performance in our entailment framework. As an example, the entailment relations between *"Apple"* and *"Macintosh"*, or between *"Iranian"* and *"IRIB"* can be represented by lexical rules which could not be extracted using WordNet or any other resource. This confirms our hypothesis that increasing the coverage

[12] http://www.cs.ualberta.ca/~lindek/downloads.htm

	VO		WN		PROX		DEP		WIKI	
	DEV	TEST	DEV	TEST	DEV	TEST	DEV	TEST	DEV	TEST
Accuracy	61.8	58.8	61.8	58.6	61.8	58.8	62.0	57.3	62.6	60.3

Table 2.2: Comparing accuracy results over the RTE-5 dataset.

and using a context sensitive approach in rule extraction, may result in a better performance in the RTE task.

Though encouraging and substantially confirming our working hypothesis, the observed performance increase is lower than expected. This might be due to the difficulty of exploiting lexical information when the tree edit distance algorithm is used, which is the basic matching algorithm employed by the EDITS package. Often, valid and reliable rules that could be potentially applied to reduce the distance between T and H are ignored because of the syntactic constraints imposed by the algorithm. To verify this hypothesis we performed another experiment, comparing the different resources in terms of potential coverage, independently from any RTE algorithm.

We performed an analysis of the coverage by the rules extracted and retained after filtering, from each resource. To this aim, we count the number of pairs in the RTE-5 data which contain lexical rules present in the WordNet, VerbOcean, Lin Dependency/Proximity, and Wikipedia repositories. For all T-H pairs of RTE-5 dataset, we computed the total rules of $w_1 \Rightarrow w_2$, that match a word w_1 in T and a word w_2 in H. Then, we estimated the number of rules that were extracted from each resource and the number of rules that were retained in our experiments. Table 2.3 shows the coverage of the content words of the extracted rules for RTE-5 from the different resources. As can be seen, the coverage of Wikipedia is the highest amongst available resources.

	VO		WN		PROX		DEP		WIKI	
	Extr.	Ret.	Extr.	Ret.	Extr.	Ret.	Extr.	Ret.	Extr.	Ret.
Coverage	0.08%	0.08%	0.4%	0.4%	3%	0.09%	2%	1%	83%	24%

Table 2.3: Coverage of rule repositories over the RTE-5 dataset.

2.3.4 Paraphrase Tables as a Source of Knowledge

In addition to extracting lexical rules from Wikipedia, we explored the usage of parallel corpora to extract paraphrase tables. Based on our definition, paraphrase tables (PPHT) contain pairs of corresponding phrases[13] in the same language, possibly associated with probabilities. They proved to be useful in a number of NLP applications such as natural language generation [Iordanskaja et al., 1991], multidocument summarization [McKeown et al., 2002], automatic evaluation of MT [Denkowski & Lavie, 2010], and TE [Dinu & Wang, 2009].

One of the proposed methods to extract paraphrases relies on a pivot-based approach using phrase alignments in a bilingual parallel corpus [Bannard & Callison-Burch, 2005]. With this method, all the different phrases in one language that are aligned with the same phrase in the other language are extracted as paraphrases. After the extraction, pruning techniques [Snover et al., 2009] can be applied to increase the precision of the extracted paraphrases.

In our work we used available paraphrase databases for English,[14] which have been extracted using the method previously outlined. Additionally, in order to experiment with different paraphrase sets providing different degrees of coverage and precision, we pruned the paraphrase table based on the probabilities, associated to its entries, of 0.1, 0.2 and 0.3. The number of phrase pairs extracted varies from 6 million to about 80,000, with an

[13] A phrase in our approach is an n-gram composed of up to 5 consecutive words.
[14] http://www.cs.cmu.edu/alavie/METEOR

Dataset	WN	VO	WIKI	PPHT	PPHT 0.1	PPHT 0.2	PPHT 0.3
RTE3	61.88	62.00	61.75	62.88	**63.38**	**63.50**	63.00
RTE5	62.17	61.67	60.00	61.33	**62.50**	**62.67**	62.33

Table 2.4: Accuracy results on RTE using different lexical resources.

average of 3.2 words per phrase.

One of the main limitations of the distance algorithms (e.g. tree edit distance) employed in EDITS package is limiting the use of lexical knowledge to only unigrams, not allowing to match the longer lexical units (e.g. phrases), with more contextual information. In order to maximize the usage of lexical knowledge, our entailment decision criterion is based on similarity scores calculated with a novel phrase-to-phrase matching process. A phrase in our approach is an n-gram composed of up to 5 consecutive words, excluding punctuation. Entailment decisions are estimated by combining phrasal matching scores calculated for each level of n-grams, which is the number of 1-grams, 2-grams,..., 5-grams extracted from H that match with n-grams in T. This algorithm is detailed in Chapter 3. To combine the phrasal matching scores obtained at each *n-gram* level, we used a Support Vector Machine classifier, SVMlight [Joachims, 1999a], using each score as a feature.

We experimented with the original RTE-3 and RTE-5 datasets, annotated with token, lemma, and stem information using the TreeTagger [Schmid, 1995] and the Snowball stemmer [Porter, 2001]. We compared the results achieved with paraphrase tables (extracted with different pruning thresholds set to 0.1, 0.2, and 0.3) with those obtained using the three most widely used English resources for Textual Entailment mentioned before.

Table 2.4 shows the accuracy results calculated over the original RTE-3 and RTE-5 test sets, training our classifier over the corresponding devel-

opment sets. The results show that pruned paraphrase tables always outperform the other lexical resources used for comparison, with an accuracy increase up to 3%. In particular, we observe that using 0.2 as a pruning threshold provides a good trade off between coverage and precision, leading to our best results on both datasets (63.50% for RTE-3, and 62.67% for RTE-5). Compared with the scores reported by participants in the two editions of the RTE Challenge (*i.e.* RTE-3 and RTE-5), these results are about 1% above the average [Mehdad et al., 2011].

Overall, these results confirm our claim that increasing the coverage using context sensitive phrase pairs obtained from large parallel corpora, results in better performance not only in RTE. We also demonstrated the effectiveness of paraphrase tables as a mean to overcome the bias toward single words featured by the existing resources mostly used by RTE systems.

2.4 Approaches

A number of approaches applied to semantics, inference and textual entailment have been experimented through the years, since the launch of the RTE Challenge in 2005. In this section, we focus on several aspects of these approaches excluding the preprocessing part, which is not in the scope of this thesis.

2.4.1 Logic-based approaches to RTE

Logic inference can be considered as one of the most direct approaches to the entailment problem. Tatu & Moldovan [2007]; Tatu et al. [2006] transformed two text snippets into three-layered semantically-rich logic form representations, generates an abundant set of lexical, syntactic, semantic, and world knowledge axioms and, iteratively, searches for a proof for the

entailment between the text T and a possibly relaxed version of the hypothesis H. They could improve the performance of their system using the lexical inference system in combination with their logical approach.

As another successful attempt in approaching approximate entailment using logical inference, Bos & Markert [2005] incorporated an automated reasoning technique in using a deep semantic analysis over T and H. In addition, they used simple shallow word overlap in combination with their logic engine to achieve high accuracy in RTE task.

MacCartney & Manning [2007] presented the first use of a computational model of natural logic for textual inference. They tried to overcome some limitations of lexical based approaches and build a more flexible and robust system than first-order logic based approaches. Their system finds a low-cost edit sequence which transforms T into H, learns to classify entailment relations across atomic edits and composes atomic entailment into a top-level entailment judgement.

Furthermore, Bar-Haim et al. [2008] created a logic-based representation of T and then performed simple inference (using WordNet and the DIRT inference rule database) over H. However, they could not show an effective method for using DIRT as an inference rules repository.

2.4.2 Transformation and similarity based approaches to RTE

The transformation-based entailment method makes use of various types of entailment knowledge to gradually transform T such that it becomes more similar to H, or vice versa. Bar-Haim et al. [2008] well-investigated this approach by exploiting different knowledge sources which were uniformly represented in the form of entailment rules. This allows to the consistent application of the same kinds of transformations on the text, regardless of the source of the knowledge. The applied transformations generate multiple consequences (new texts entailed from the original one), whose parse trees

are efficiently stored in a packed representation, termed compact forest. An approximate matching phase makes the final entailment decision by assessing the degree of syntactic match between the hypothesis and the generated consequents, compensating for knowledge gaps in the available rules.

Kouylekov & Magnini [2005] assumed a distance-based framework, where the distance between T and H is inversely proportional to the entailment relation in the pair, estimated as the sum of the costs of the edit operations (*i.e.* insertion, deletion, substitution) on the parse tree, which are necessary to transform T into H. They use different resources to estimate the edit operations cost and to ensure the non-symmetric directionality of the entailment relation.

Moreover, they developed the first open source system for RTE which implements a collection of algorithms, providing a configurable framework to quickly set up a working environment to experiment with the RTE task [Kouylekov & Negri, 2010]. As a novel part of this thesis, we proposed a method to estimate and optimize the operation costs in distance-based approaches, applying the Particle Swarm Optimization algorithm [Mehdad, 2009; Mehdad & Magnini, 2009a]. Moreover, we implemented an innovative method to assess the results achieved by EDITS on a given training corpus in [Kouylekov et al., 2011]. Note that these methods are descried in Section 2.5 as contributions of this PhD thesis.

Furthermore, Harmeling [2009] introduced a system for textual entailment that is based on a probabilistic model of entailment. This model is defined using a calculus of transformations on dependency trees, where derivations in that calculus preserve the truth, only with a certain probability.

In the RTE scenario, since T is often much longer than H, if the surface string of H is very similar to a part of T, this is an indication that H might

be entailed by T. Malakasiotis [2009] compared H to a sliding window of T's string of the same size by calculating the largest similarity score to estimate whether T entails H or not. They also exploited WordNet to detect synonyms, and a dependency parser to measure similarity in the grammatical structure of T and H in order to enrich their feature space.

In order to boost the similarity scores and extend them to a different level, Iftene & Balahur-Dobrescu [2007]; Iftene & Moruz [2009] compared H's parse tree against subtrees of T's parse tree. They transformed the hypothesis making use of an extensive semantic knowledge from sources like DIRT, WordNet, Wikipedia, and acronyms database. Additionally, they took advantage of hand coded complex grammar rules for rephrasing in English. Besides, some systems exploited different aligning and matching algorithms at the lexical (*e.g.* [Glickman et al., 2006]), phrase (*e.g.* [Padó et al., 2008]), syntactic (*e.g.* [Yatbaz, 2008]), semantic (*e.g.* [Li et al., 2009b]), or onthology (*e.g.* [Siblini & Kosseim, 2008]) level. In addition, Sammons et al. [2009] proposed an architecture designed to integrate different and unscaled natural language processing resources, combining them taking advantage of an alignment-based method.

In measuring the similarity at different levels, syntactic or semantic representations of the input expressions cannot always be estimated accurately (*e.g.*, due to parser errors). For this reason, the methods that operate at the syntactic or semantic level do not necessarily outperform the methods that operate on surface strings [Wang, 2011]. Another problem in approaching TE with similarity metrics is that the entailment relation is an asymmetric relation, while most of the similarity relations are symmetric.

2.4.3 Supervised Learning Methods for RTE

Inside the different approaches to TE, the use of Machine Learning (ML) approaches is dominant. This is mainly because both logic and rule-based

methods suffer from either limited coverage of hand-crafted rules and lower performance.

In ML approaches, a variety of features including lexical, syntactic and semantic features can be extracted from training examples, thus can be employed to train a classifier. For instance, Agichtein et al. [2008] used a supervised machine learning approach to train a classifier over a variety of lexical, syntactic, and semantic metrics. They treated the output of each metric as a feature, and train a classifier on the provided data from the available RTE datasets. In the same direction, Rodrigo et al. [2008] extracted syntactic and semantic features after applying dependency parsing and NE recognition, while Nielsen et al. [2009] and Bensley & Hickl [2008] focused on collecting deeper semantic features.

The approach proposed in [Wang & Neumann, 2008a] is based on constructing structural features from the abstract tree descriptions, which are automatically extracted from syntactic dependency trees of T and H. These features are then applied by a subsequence-kernel-based classifier that learns to decide whether the entailment relation holds between two texts. A divide-and-conquer architecture is then in charge of providing a set of specific RTE methods (namely: temporal anchors, named entities and noun phrase anchors), and then combine them applying a voting scheme in order to maximize the accuracy.

The system described in [Zanzotto & Moschitti, 2006a] defines a cross-pair similarity measure based on the syntactic trees of T and H, and combines such similarity with traditional intra-pair similarities to define a novel semantic kernel function. The intuition behind this approach is that not only intra-pair similarity between T and H, but also cross-pair similarity between two pairs can be useful to address the problem. The latter similarity measure along with a set of annotated examples is used by a learning algorithm to automatically derive syntactic and lexical rules to

solve complex entailment cases.

In this dissertation, inline with moving beyond the state-of-the-art in RTE systems, we also describe our novel approach applying tree kernels [Collins & Duffy, 2002]. We proposed models for effectively using syntactic and semantic information in RTE, without requiring either large automatic rule acquisition or hand-coding. These models exploit lexical similarities to generalize lexical-syntactic rules automatically derived by supervised learning methods. The joint syntactic/semantic model is realized by means of novel tree kernels, which can match sub-trees whose leaves are lexically similar (not just identical). This approach and the related results described in Section 2.6.

2.5 Optimizing Edit Distance based Entailment

In this section, we introduce the need of optimization for edit distance based TE approaches (*e.g.* [Kouylekov & Magnini, 2005] and [Kouylekov & Negri, 2010]). We firstly discuss the notion of the problem as well as the motivation of our approach in optimizing the cost of edit operations in edit distance based techniques. Then, we propose and describe our solution followed by experimental results. We also show the need for an automatic way to explore the large search space of possible configurations, in order to select the most promising one for a given RTE dataset. Finally, we explain our proposed solution using optimization techniques and comment the results we achieved on all previous RTE datasets.

2.5.1 Optimizing Edit Distance Using Particle Swarm Optimization

Among the approaches to the problem of textual entailment discussed in Section 2.4, some methods use the notion of distance between the pair of T

and H as the main feature which separates the entailment classes (positive and negative). Some systems calculate the distance by implementing Tree Edit Distance (TED), based on the syntactic features that are represented in the structured parse tree of each string [Kouylekov & Magnini, 2005]. In this method the distance is computed as the cost of the edit operations (insertion, deletion and substitution) that transform the text T into the hypothesis H. Each edit operation has an associated cost and the entailment score is calculated such that the set of operations would lead to the minimum cost.

Generally, the initial cost is assigned to each edit operation empirically, or based on the expert knowledge and experience. These methods arise a critical problem when the domain, field or application is new and the level of expertise and empirical knowledge is very limited. In dealing with textual entailment, Kouylekov & Magnini [2006] tried to experiment different cost values based on various linguistic knowledge and probabilistic estimations. For instance, they defined the substitution cost as a function of similarity between two nodes, or, for the insertion cost, they employed Inverse Document Frequency (IDF) of the inserted node. However, the results were not optimal.

Other approaches towards estimating the cost of operations in TED tried to learn a generic or discriminative probabilistic model from the data [Bernard et al., 2008; Neuhaus & Bunke, 2004], without concerning the optimal value of each operation. One of the drawbacks of those approaches is that the cost values of edit operations are hidden behind the probabilistic model. Additionally, the cost can not be weighted or varied according to the tree context and node location [Bernard et al., 2008].

In order to overcome these drawbacks, we propose a stochastic method based on Particle Swarm Optimization (PSO) to estimate the cost of each edit operation for TE problem. By integrating PSO, we try to learn the

optimal cost for each operation in order to improve the prior textual entailment model. Our innovative contribution is to automatically estimate the best possible operation costs on the development set. A further advantage of such method, besides automatic learning of the operation costs, is being able to investigate the cost values to better understand how TED approaches the textual entailment datasets.

Particle Swarm Optimization (PSO)

PSO is a stochastic optimization technique which takes inspiration from the social behavior of bird flocking and fish schooling [Eberhart et al., 2001]. PSO is one of the population-based search methods which takes advantage of the concept of social sharing of information. In this algorithm each particle can learn from the experience of other particles in the same population (called swarm). In other words, each particle in the iterative search process would adjust its flying velocity as well as position not only based on its own acquaintance but also other particles' flying experience in the swarm. This algorithm has found efficient in solving a number of engineering problems. PSO is mainly built on the following equations.

$$X_i = X_i + V_i \tag{2.1}$$

$$V_i = \omega V_i + c_1 r_1 (X_{bi} - X_i) + c_2 r_2 (X_{gi} - X_i) \tag{2.2}$$

To be concise, for each particle at each iteration, the position X_i (Equation 2.1) and velocity V_i (Equation 2.2) is updated. X_{bi} is the best position of the particle during its past routes and X_{gi} is the best global position over all routes travelled by the particles of the swarm. r_1 and r_2 are random variables drawn from a uniform distribution in the range [0,1], while c_1 and c_2 are two acceleration constants regulating the relative velocities with respect to the best local and global positions. The weight ω is used

as a trade-off between the global and local best positions. It is usually selected slightly less than 1 for better global exploration [Melgani & Bazi, 2008]. The best position is computed based on the fitness function defined in association with the related problem. Both position and velocity are updated during the iterations until convergence is reached or iterations attain the maximum number defined by the user.

Integrating TED with PSO

This section aims at finding the optimal set of operation costs to: *i)* improve the performance of TED in different applications, and *ii)* provide some information on how different operations in TED approach an application or dataset.

One of the most important steps in applying PSO is to define a fitness function which could lead the swarm to the optimized particles in different applications and over different datasets. The choice of this function is very crucial, since PSO evaluates the quality of each candidate particle for driving the solution space to optimization, on the basis of the fitness function. Moreover, this function should possibly improve the textual entailment recognition model.

In order to attain these goals, we tried to define accuracy obtained from a TED based system as a good fitness function in optimizing the cost values. Since maximizing the accuracy would directly increase the performance of the system or enhance the model to solve the problem, this measure is a possible choice to adapt in order to achieve our aim. In this method, trying to maximize the fitness function will compute the best model based on the optimal cost values in the particle space of PSO algorithm.

The procedure describing the proposed system to optimize and estimate the cost of edit operations in TED applying PSO algorithm is as follows.

a) *Initialization*

Step 1) Generate a random swarm of particles (in a simple case each particle is defined by the cost of three operations).

Step 2) For each position of the particle from the swarm, obtain the fitness function value (accuracy) over the training data.

Step 3) Set the best position of each particle with its initial position (X_{bi}).

b) *Search*

Step 4) Detect the best global position (X_{gi}) in the swarm based on maximum value of the fitness function over all explored routes.

Step 5) Update the velocity of each particle (V_i).

Step 6) Update the position of each particle (X_i). In this step, by defining the boundaries, we could stop the particle to exit the allowed search space.

Step 7) For each candidate particle calculate the fitness function (accuracy).

Step 8) Update the best position of each particle if the current position has a larger value.

c) *Convergence*

Step 9) Run till the maximum number of iteration (in our case set to 10) is reached or start the search process.

d) *Results*

Step 10) Return the best fitness function value and the best particle. In this step the optimum costs are returned.

Following the steps above, in contrary to determine the entailment relation applying tree edit distance, the operation costs can be automatically

estimated and optimized. In this process, both fitness functions could be easily compared and the cost values leading to the better model would be selected.

Experiments

In our experiments, in order to deal with TED approach to textual entailment, we used the EDITS package (Edit Distance Textual Entailment Suite) [Kouylekov & Negri, 2010; Negri et al., 2009], an open source software based on edit distance algorithms which computes the T-H distance as the cost of the edit operations (*i.e.* insertion, deletion and substitution) that are necessary to transform T into H. By defining the edit distance algorithm and a cost scheme (assigning a cost to the edit operations), this package is able to learn a TED threshold, over a set of string pairs, to decide if the entailment exists in a pair. In addition, we exploited the JSwarm-PSO package [Cingolani, 2005], with some adaptations, as an implementation of the PSO algorithm.

Our experiments were conducted on the RTE datasets.[15] Each pair in the datasets was enriched with two syntactic dependency parse trees using the Stanford statistical parser [Klein & Manning, 2003]. The accuracy, by default, is computed by EDITS over the training set based on 10-fold cross-validation.

We conducted six different experiments on each RTE-1 to RTE-4 dataset.[16] The costs were estimated on the training set and the results obtained based on the estimated costs over the test set.

In the first set of experiments, we set a simple cost scheme based on three operations. Implementing this cost scheme, we expect to optimize the cost of each edit operation without considering that the operation costs

[15] http://www.pascal-network.org/Challenges/RTE1-4

[16] At the time of experiments, the only available dataset were RTE-1 to RTE-4)

may vary based on different characteristics of a node, such as size, location or content. The results were obtained considering three different settings: *i)* a random cost assignment, *ii)* assigning the cost based on the human expertise knowledge and intuition (called Intuitive), and *iii)* automatic estimated and optimized cost for each operation. In the second case, we used the same scheme which was by EDITS expert users and developers.

In the second set of experiments, we tried to compose an advanced cost scheme with more fine-grained operations to assign a weight to the edit operations based on the characteristics of the nodes. For example if a node is in the list of stop-words, the deletion cost is set to zero. Otherwise, the cost of deletion would be equal to the number of words in H multiplied by word's length (number of characters). Similarly, the cost of inserting a word w in H is set to 0 if w is a stop word, and to the number of words in T multiplied by word's length otherwise. The cost of substituting two words is the Levenshtein distance (*i.e.* the edit distance calculated at the level of characters) between their lemmas, multiplied by the number of words in T, plus number of words in H. By this intuition, we tried to optimize nine specialized costs for edit operations (*i.e.* each particle is defined by 9 parameters to be optimized). We conducted the experiments using all three cases mentioned in the simple cost scheme.

In each experiment, we applied both fitness functions in the optimization. However, at the final phase, the costs which led to the maximum results were chosen as the estimated operation costs. In order to save time, we set the number of iterations to 10, in addition, the weight ω was set to 0.95 for better global exploration [Melgani & Bazi, 2008].

Results

Our results are summarized in Table 2.5. We show the accuracy gained by a distance-based (word-overlap) baseline for textual entailment [Mehdad

& Magnini, 2009b] to be compared with the results achieved by the random, intuitive and optimized cost schemes using EDITS system. For the better comparison, we also present the results of the EDITS system in RTE-4 challenge using a combination of different distances as features for classification [Cabrio et al., 2008].

In the first experiment, we estimated the cost of each operation using the simple cost scheme. Table 2.5 shows that in all datasets, accuracy improved up to 9% by optimizing the cost of each edit operation. Results prove that the optimized cost scheme enhances the quality of the system performance, even more than the cost scheme used by experts (Intuitive cost scheme).

Furthermore, in the second set of experiments, using the fine-grained and weighted cost scheme for edit operations we could achieve the highest results in accuracy. Achieved results illustrate that all optimized results outperform the word-overlap baseline for textual entailment as well as the accuracy obtained in RTE-4 challenge using combination of different distances as features for classification.

By exploring the estimated optimal cost of each operation, another interesting point was discovered. The estimated cost of deletion in the first set of experiments was 0, which means that deleting a node from the dependency tree of T does not effect the quality of results. This proves that by setting different cost schemes, we could explore even some linguistics phenomena which exists in the entailment dataset. Studying the dataset from this point of view might be interesting to find some hidden information which can not be explored easily.

In addition, the optimized model can reflect more consistency and stability (from 59% to 62% in accuracy) than other models, while in unoptimized models the result varies more, on different datasets (from 50% in RTE-1 to 59% in RTE-3).

Model		Data set RTE-4	RTE-3	RTE-2	RTE-1
Simple	Random	49.6	53.6	50.4	50.5
	Intuitive	51.3	59.6	56.5	49.8
	Optimized	**56.5**	**61.6**	**58.0**	**58.1**
Advanced	Random	53.60	52.0	54.6	53.5
	Intuitive	57.6	59.4	57.7	55.5
	Optimized	**59.5**	**62.4**	**59.9**	**58.6**
Baseline		55.2	60.9	54.8	51.4
RTE-4 Challenge		57.0			

Table 2.5: Comparison of accuracy on RTE datasets based on optimized and unoptimized cost schemes.

2.5.2 Optimizing Textual Entailment Recognition System Using Genetic Algorithm

Generally, it would be useful for RTE system developers to have: *i)* automatic ways to support systems' tuning at a training stage, and *ii)* reliable terms of comparison to validate their hypotheses, and position the results of their work before submitting runs for evaluation. In this section we address these needs by extending an open-source RTE package with a mechanism that automatizes the selection of the most promising configuration over a training dataset.

EDITS is an open source package for recognizing textual entailment, which offers a modular, flexible, and adaptable working environment to experiment with the RTE task over different datasets. The package allows to: *i)* create an entailment engine by defining its basic components (*i.e.* algorithms, cost schemes, rules, and optimizers); *ii)* train such entailment engine over an annotated RTE corpus to learn a model; and *iii)* use the entailment engine and the model to assign an entailment judgment and

a confidence score to each pair of the test corpus. A key feature of EDITS is represented by its high configurability, allowed by the availability of different algorithms, the possibility to integrate different sets of lexical entailment and contradiction rules, and the variety of parameters for performance optimization (as it was discussed in Section 2.5.1).

Although configurability is *per se* an important aspect (especially for an open-source and general purpose system), there is another side of the coin. In principle, in order to select the most promising configuration over a given development set, one should exhaustively run a huge number of training/evaluation routines. Such number corresponds to the total number of configurations allowed by the system, which result from the possible combinations of parameter settings. When dealing with growing dataset sizes, and the tight time constraints usually posed by the evaluation campaigns, this problem becomes particularly challenging, as developers are hardly able to run exhaustive training/evaluation routines. Such situation results in running a limited number of experiments with the most "reasonable" configurations, which consequently might not lead to the optimal solution.

The need of a mechanism to automatically obtain the most promising solution on one side, and the need of efficiency on the other side, arise the necessity to optimize this procedure. Along this direction, the objective is good a trade-off between exhaustive experimentation with all possible configurations (infeasible), and educated guessing (unreliable). The remainder of this section tackles this issue introducing an optimization strategy based on genetic algorithms, another optimization algorithm that match our optimization criteria, and describing its adaptation to extend EDITS with the new functionality.

Genetic Algorithm (GA)

Genetic algorithms (GA) are well suited to efficiently deal with large search spaces, and have been recently applied with success to a variety of optimization problems and specific NLP tasks [Figueroa & Neumann, 2008; Rodríguez et al., 2008]. GA are a direct stochastic method for global search and optimization, which mimics natural evolution. To this aim, they work with a *population of individuals*, representing possible solutions to the given task. Traditionally, solutions are represented in binary as strings of *0*s and *1*s, but other encodings (*e.g.* sequences of real values) are possible. The evolution usually starts from a population of randomly generated individuals, and at each generation selects the best suited individuals based on a *fitness function* (which measures the optimality of the solution obtained by the individual). Such selection is then followed by *modifications* of the selected individuals obtained by recombining (crossover) and performing random changes (mutation) to form a new population, which will be used in the next iteration. Finally, the algorithm is terminated when the maximum number of generations, or a satisfactory fitness level has been reached for the population.

Integrating EDITS with Genetic Algorithm

Our extension to the EDITS package, integrating with GA (EDITS-GA), consists in an iterative process that starts with an initial population of randomly generated configurations. After a training phase with the generated configurations, the process is evaluated by means of the fitness function, which is manually defined by the user.[17] This measure is used by the genetic algorithm to iteratively build new populations of configurations, which are trained and evaluated.

[17]For instance, working on the RTE Challenge "Main" task data, the fitness function would be the *accuracy* for RTE1 to RTE5.

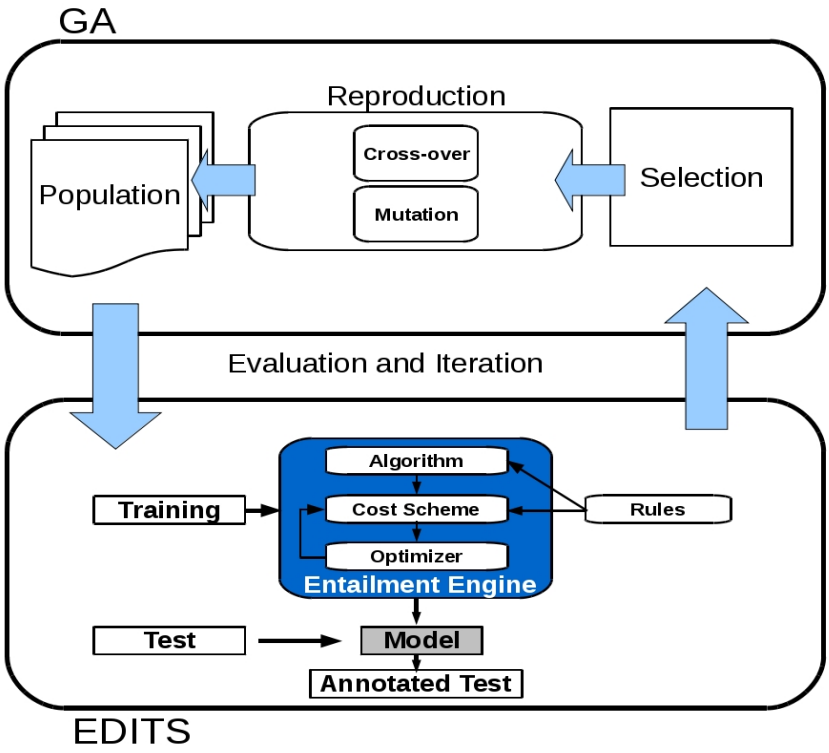

Figure 2.1: EDITS-GA framework.

This process can be seen as the combination of: *i)* a micro training/evaluation routine for each generated configuration of the entailment engine; and *ii)* a macro evolutionary cycle, as illustrated in Figure 2.1. The fitness function is an important factor for the evaluation and the evolution of the generated configurations, as it drives the evolutionary process by determining the best-suited individuals used to generate new populations. The procedure to estimate and optimize the best configuration applying the GA, can be summarized as follows.

(1) *Initialization*: generate a random initial population (*i.e.* a set of configurations).
(2) *Selection*:

2a. The fitness function (*e.g.* accuracy, or F-measure) is evaluated for each individual in the population.

2b. The individuals are selected according to their fitness function value.
(3) *Reproduction*: generate a new population of configurations from the selected one, through genetic operators (cross-over and mutation).
(4) *Iteration*: repeat the *Selection* and *Reproduction* until *Termination*.
(5) *Termination*: end if the maximum number of iterations has been reached, or the population has converged towards a particular solution.

It is worth to mention that, due to the nature of GAs, the iterative evolutionary process does not explore the entire search space, and is not guaranteed to converge to the best individual solution.

Experiments

Our experiments were carried out over the datasets used in the six editions of the RTE Challenge ("Main" task data from RTE1 to RTE6). For each dataset we obtained the best model by training EDITS-GA over the development set, and evaluating the resulting model on the test pairs. To this aim, the optimization process is iterated over all the available algorithms in order to select the best combination of parameters. As *termination* criterion, we set to 20 the maximum number of iterations.

In order to extend EDITS with genetic algorithms, we used a GA implementation available in the JGAP tool.[18] In our settings, each individual contains a sequence of boolean parameters corresponding to the activation/de-activation of the system's basic components (algorithms, cost schemes, rules, and optimizers). The configurations corresponding to such individuals constitute the populations iteratively evaluated by EDITS-GA on a given dataset.

To increase efficiency, we extended EDITS to pre-process each dataset using the tokenizer and stemmer available in Lucene.[19] This pre-processing

[18] http://jgap.sourceforge.net/
[19] http://lucene.apache.org/

phase is automatically activated when the EDITS-GA has to process non-annotated datasets. However, we also annotated the RTE corpora with the Stanford parser plug-in (downloadable from the EDITS website[20]) in order to run the syntax-based algorithms available (*e.g.* tree edit distance).

The number of boolean parameters used to generate the configurations is 18. In light of this figure, it becomes evident that the number of possible configurations is too large (2^{18}=262,144) for an exhaustive training/evaluation routine over each dataset.[21] However, with an average of 5 *reproductions* on each iteration, EDITS-GA makes an average of 100 configurations for each algorithm. Thanks to EDITS-GA, the average number of evaluated configurations for a single dataset is reduced to around 400.[22]

Results

Our results are summarized in Table 2.6, showing the highest, lowest, and average score achieved by participants in the RTE challenges. Moreover, the official results obtained by EDITS are compared with the performance achieved with EDITS-GA on the same data.[23] We can observe that, for all datasets, the results achieved by EDITS-GA significantly improve (up to 4.51%) the official EDITS results. It's also worth mentioning that such scores are always higher than the average ones obtained by participants.

This confirms that EDITS-GA can be potentially used by RTE systems developers as a strong term of comparison to assess the capabilities of their own system. Since time is a crucial factor for RTE systems, it is important to remark that EDITS-GA allows to converge on a promising configuration

[20] http://edits.sf.net/

[21] In an exploratory experiment we measured in around **4 days** the time required to train EDITS, with all possible configurations, over small datasets (RTE1 to RTE5). All time figures are calculated on an Intel(R) Xeon(R), CPU X3440 @ 2.53GHz, 8 cores with 8 GB RAM.

[22] With these settings, training EDITS-GA over small datasets (RTE1 to RTE5) takes about **9 minutes** each, calculated on an Intel(R) Xeon(R), CPU X3440 @ 2.53GHz, 8 cores with 8 GB RAM.

[23] As regards RTE-3, EDITS was not among the participating systems.

	Best	Lowest	Average	EDITS (rank)	EDITS-GA (rank)	% Impr.	Comp. Time
RTE1	0.586	0.495	0.544	0.559 (8)	**0.5787** (3)	+3.52%	8m 24s
RTE2	0.7538	0.5288	0.5977	0.605 (6)	**0.6225** (5)	+2.89%	9m 8s
RTE3	0.8	0.4963	0.6237	-	**0.6875** (4)	-	9m
RTE4	0.746	0.516	0.5935	0.57 (17)	**0.595** (10)	+4.38%	30m 54s
RTE5	0.735	0.5	0.6141	0.6017 (14)	**0.6233** (9)	+3.58%	8m 23s
RTE6	0.4801	0.116	0.323	0.4471 (4)	**0.4673** (3)	+4.51%	1h 54m 20s

Table 2.6: RTE results (acc. for RTE1-RTE5, F-meas. for RTE6).

quite efficiently.

As can be seen in Table 2.6, the whole process takes around 9 minutes for the smaller datasets (RTE1 to RTE5), and less than 2 hours for a very large dataset (RTE6). Such time analysis further proves the effectiveness of the extended EDITS-GA framework.

For the sake of completeness we studied the differences between the "educated guessing" done by the EDITS developers for the official RTE submissions, and the "optimal" configuration automatically selected by EDITS-GA. Surprisingly, in some cases, even a minor difference in the selected parameters leads to significant gaps in the results. For instance, in RTE-6 dataset, the "guessed" configuration [Kouylekov et al., 2010b] was based on the lexical overlap algorithm, setting the cost of replacing H terms without an equivalent in T to the minimal Levenshtein distance between such words and any word in T. EDITS-GA estimated, as a more promising solution, a combination of lexical overlap with a different cost scheme (based on the IDF of the terms in T). In addition, in contrast with the "guessed" configuration, stop-words filtering was selected as an option, eventually leading to a 4.51% improvement over the official RTE6 result.

2.6 Syntactic Semantic Learning for Textual Entailment Recognition

For all methods discussed in Section 2.4, the effective use of syntactic and semantic information depends on the coverage and the quality of the specific rules. Lexical and syntactic rules can be automatically extracted from plain corpora but the quality (also in terms of noise) and the coverage is low. In contrast, rules written at the semantic level are more accurate but their automatic design is difficult and so they are typically hand coded for the specific phenomena.

In this section, we propose models for effectively using syntactic and semantic information in RTE, without requiring either large automatic rule acquisition or hand-coding. These models exploit lexical similarities to generalize lexical-syntactic rules automatically derived by supervised learning methods. In more detail, syntax is encoded in the form of parse trees whereas similarities are defined by means of WordNet similarity measures or Latent Semantic Analysis (LSA) applied to Wikipedia or to the British National Corpus (BNC). The joint syntactic/semantic model is realized by means of novel tree kernels, which can match subtrees whose leaves are lexically similar or related (not just identical).

2.6.1 Motivating Example

Lexical and syntactic rules are largely used in textual entailment recognition systems (reported in Section 2.3) as they conveniently encode world knowledge into linguistic structures. For example, in:

$T_2 \Rightarrow ? H_2$	
T_2	"*In 1980 Chapman killed Lennon.*"
H_2	"*John Lennon died in 1980.*"

to decide whether the simple sentences are in the entailment relation, we need a lexical-syntactic rule such as:

$$\rho_1 = \mathbf{X} \text{ killed } \mathbf{Y} \rightarrow \mathbf{Y} \text{ died}$$

along with such rules, the temporal information should be taken into consideration.

Supervised approaches were experimented in [Zanzotto & Moschitti, 2006b; Zanzotto et al., 2009], where lexical-syntactic rules were derived from examples in terms of complex relational features. This approach can easily miss some useful information and rules. Given the pair $\langle T_2, H_2 \rangle$, to derive the entailment value of the following case:

$T_3 \Rightarrow ? H_3$	
T_3	"*In 1963 Lee Harvey Oswald murdered JFK*"
H_3	"*JFK died in 1963*"

we can only rely on this relatively interesting lexical-syntactic rule (*i.e.* which is in common between the two examples):

$$\rho_2 = (\text{VP } (\text{VBZ}) \ (\text{NP } \mathbf{X})) \rightarrow (\text{S } (\text{NP } \mathbf{X})(\text{VP } (\text{VBZ} \quad \text{died})))$$

Unfortunately, this can be extremely misleading since it also derives similar decisions for the following example:

$T_4 \Rightarrow ? H_4$	
T_4	"*In 1956 JFK met Marilyn Monroe*"
H_4	"*Marilyn Monroe died in 1956*"

The problem is that the pairs $\langle T_2, H_2 \rangle$ and $\langle T_3, H_3 \rangle$ share more meaningful features than the rule 2, which should make the difference with respect to the relation between the pairs $\langle T_2, H_2 \rangle$ and $\langle T_4, H_4 \rangle$. Indeed, the word

kill is more semantically related to *murdered* than to *meet*. Using this information, it is possible to derive more effective rules from training examples.

There are several solutions for taking this information into account, *e.g.* by using FrameNet semantics (*e.g.*, like in [Burchardt et al., 2007]), it is possible to encode a lexical-syntactic rule using the KILLING and the DEATH frames, *i.e.*:

$$\rho_3 = \text{KILLING(Killer:}\mathbf{X}\text{,Victim:}\mathbf{Y}\text{)} \rightarrow \text{DEATH(Protagonist:}\mathbf{Y}\text{)}$$

However, to use this model, specific rules and a semantic role labeler on the specific corpora are needed. In the following sections we describes lexical similarity approaches, which can serve the generalization purpose, and also we explain how to integrate lexical similarity in syntactic structures using syntactic/semantic tree kernels for RTE.

2.6.2 Lexical similarities

As it was discussed in Sections 2.3 and 2.4, in RTE many lexical similarity measures based on different resources or corpora has been used. For example, WordNet similarities [Pedersen et al., 2004], or Latent Semantic Analysis over a large corpus, are widely used in many systems and approaches (*e.g.* [Kouylekov et al., 2010a]).

In this section we present the main component of our new kernel, *i.e.* a lexical similarity derived from different resources. This is used inside the syntactic/semantic tree kernel to enhance the basic tree kernel functions.

WordNet Similarities have been heavily used in previous NLP work. All WordNet similarities apply to pairs of synonymy sets (synsets) and return a value indicating their semantic relatedness. For example, the

following measures, that we use in our study, are based on path lengths between concepts in the Wordnet Hierarchy:

Path : this measure is equal to the inverse of the shortest path length ($path\_length$) between two synsets c_1 and c_2 in WordNet

$$Sim_{Path}(w_1, w_2) = \frac{1}{path\_length(c_1, c_2)} \tag{2.3}$$

WUP : the Wu and Palmer [Wu & Palmer, 1994] similarity metric is based on the depth of two given synsets c_1 and c_2 in the WordNet taxonomy, and the depth of their least common subsumer (lcs). These are combined into a similarity score:

$$Sim_{WUP}(w_1, w_2) = \frac{2 \times depth(lcs)}{depth(c_1) + depth(c_2)} \tag{2.4}$$

Wordnet similarity measures on synsets can be extended to similarity measures between words as follows:

$$\kappa_{\mathcal{S}}(w_1, w_2) = max_{(c_1,c_2)\in C_1\times C_2} Sim_{\mathcal{S}}(c_1, c_2) \tag{2.5}$$

where $\mathcal{S}$ is Path or WUP and C_i is the set of the synsets related to the word w_i.

Distributional Semantic Similarity, based on Latent Semantic Analysis (LSA), is one of the corpus-based measure of distributional semantic similarity [Landauer et al., 1998]. In this method, words are represented in a document space as features vectors (*i.e.* $\vec{w_i}$). Each feature is a document and its value is the frequency of the word in the document. The similarity is generally computed as a cosine similarity:

$$\kappa_{LSI}(w_1, w_2) = \frac{\vec{w_1}\vec{w_2}}{|\vec{w_1}||\vec{w_2}|} \tag{2.6}$$

In our approach we define a proximity matrix P where $p_{i,j}$ represents $\kappa_{LSI}(w_i, w_j)$. The core of our approach lies on LSI (Latent Semantic Indexing) over a large corpus. We used singular value decomposition (SVD) to build the proximity matrix $P = DD^T$ from a large corpus, represented by its word-by-document matrix D.

SVD decomposes D (weighted matrix of term frequencies in a collection of texts) into three matrices $U\Sigma V^T$, where U (matrix of term vectors) and V (matrix of document vectors) are orthogonal matrices whose columns are the eigenvectors of DD^T and D^TD respectively, and Σ is the diagonal matrix containing the singular value of D.

Given such decomposition, P can be obtained as $U_k\Sigma_k^2U_k^T$, where U_k is the matrix containing the first k columns of U and k is the dimensionality of the latent semantic space. This is used to efficiently reduce the memory requirements while retaining the information. Finally we computed the term similarity using the cosine measure in the vector space model.

Generally, LSA can be observed as a way to overcome some of the drawbacks of the standard vector space model, such as sparseness and dimensionality. Put it in a different way, the LSA similarity is computed in a lower dimensional space, in which second-order relations among words and documents are exploited [Mihalcea et al., 2006].

It is worth mentioning that the LSA similarity measure depends on the selected corpus but it benefits from a higher computation speed in comparison to the construction of the similarity matrix based on the WordNet Similarity package [Pedersen et al., 2004].

2.6.3 Integrating Semantic in Syntactic Tree Kernels

In Section 2.4 we have shown that the role of the syntax for RTE is important but it is not enough. Therefore, the lexical similarity described in the previous section should be taken into account in the model definition.

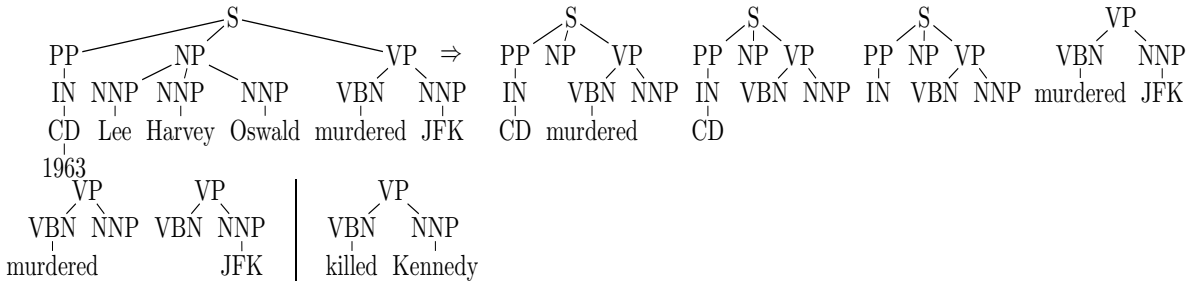

Figure 2.2: A syntactic parse tree (on the left) along with some of its fragments. After the bar there is an important fragment from a semantically similar sentence, which cannot be matched by STK but it is matched by SSTK.

Since tree kernels have been shown to be very effective for exploiting syntactic information in natural language tasks, a promising idea is to merge together the two different approaches, *i.e.* tree kernels and semantic similarities.

Syntactic Tree Kernel (STK) computes the number of common substructures between two trees T_1 and T_2 without explicitly considering the whole fragment space. The standard definition of the STK, given in [Collins & Duffy, 2002], allows for any set of nodes linked by one or more entire production rules to be valid substructures. The formal characterization is given in [Collins & Duffy, 2002], so we omit to bring it here.

Figure 2.2 shows some fragments (out of the overall 472) of the syntactic parse tree on the left, which is derived from the text T4. These fragments satisfy the constraint that grammatical rules cannot be broken. For example, *(VP (VBN (murdered) NNP (JFK)))* is a valid fragment whereas *(VP (VBN (murdered))* is not. One drawback of such kernel is that two sentences expressing similar semantics but with different lexicals produce structures

which will not be matched. For example, after the vertical bar there is a fragment, extracted from the parse tree of a semantically identically sentences: "`In 1963 Oswald killed Kennedy`". In this case, much less matches will be counted by the kernel function applied to such parse trees and the one of T4. In particular, the VP subtrees will not be matched.

To tackle this problem the Syntactic Semantic Tree Kernel (SSTK) was defined in [Bloehdorn & Moschitti, 2007].

Syntactic Semantic Tree kernels (SSTK) produces the same matches as STK. Moreover, the fragments, which are identical but for their lexical nodes, produce a match proportional to the product of the similarity between their corresponding words. Indeed, since the structures are the same, each word in position i of the first fragment is associated with a word in the same position i in the second fragment. More formally, we provide a fast evaluation of the semantic Δ function, which is identical to the one of STK plus the following step:

0. if n_1 and n_2 are pre-terminals and $label(n_1) = label(n_2)$ then $\Delta(n_1, n_2) = \lambda\kappa_{\mathcal{S}}(ch^1_{n_1}, ch^1_{n_2})$

Where $label(n_i)$ is the label of node n_i and $\kappa_{\mathcal{S}}$ is a term similarity kernel, *e.g.* based on Wikipedia, Wordnet or BNC, defined in Section 2.6.2. Note that since n_1 and n_2 are pre-terminals of a parse tree they can have only one child (*i.e.* $ch^1_{n_1}$ and $ch^1_{n_2}$) and such children are words.

For example, the fragments: *(VP (VBN (murdered) NNP (JFK)))* and *(VP (VBN (killed) NNP (Kennedy)))* will give the contribution of $\kappa_{\mathcal{S}}(murdered, kill) \times \kappa_{\mathcal{S}}(JFK, Kennedy)$ to SSTK, where $\kappa_{\mathcal{S}}$ is a lexical similarity.

Beside the novelty of taking into account tree fragments that are not identical it should be noted that the lexical semantic similarity is constrained in syntactic structures, which limit errors/noise due to incorrect

(or, as in our case, not provided) word sense disambiguation.

Finally, it should be noted that when a valid kernel is used in place of $\kappa_{\mathcal{S}}$, SSTK is a valid kernel for definition of convolution kernels [Haussler, 1999]. Since the matrix P derived by applying LSA produces a semi-definite matrix (see [Cristianini & Holloway, 2001]) we can always use the similarity matrix derived by LSA in SSTK. In case of Wordnet, the validity of the kernel will depend of the kind of similarity used. In our experiments, we have carried out single value decomposition and we have verified that our Wordenet matrices, Path and WUP, are indeed positive semi-definite.

2.6.4 Semantic Syntactic Tree Kernels for RTE

In this section, we describe how we use the syntactic tree kernel (STK) and the semantic/syntactic tree kernel (SSTK) for modeling lexical-syntactic kernels for textual entailment recognition. We build on the kernel described in [Zanzotto & Moschitti, 2006b; Zanzotto et al., 2009] that can model lexical-syntactic rules with variables (*i.e.* first-order rules).

Anchoring and pruning: Kernels for modeling lexical-syntactic rules with variables presuppose that words in texts T are explicitly related to words in hypotheses H. This correlation is generally called anchoring and it is implemented with placeholders that co-index the syntactic trees derived from T and H. Words and intermediate nodes are co-indexed when equal or similar. For example, in the pair:

$T_5 \Rightarrow ? H_5$

T_5	"*Lee Harvey Oswald was born in New Orleans, Louisiana, and was of English, German, French and Irish ancestry. In* 1963_1 *Oswald murdered* JFK_2"
H_5	"JFK_1 *died in* 1963_1"

Moreover, the set of anchors also allow us to prune fragments of the text T that are irrelevant for the final decision: we can discard sentences or phrases uncovered by place-holders. For example, in the pair $\langle T_5, H_5 \rangle$, we can infer that "*Lee H. . . ancestry*" is not a relevant fragment and remove it. This allows us to focus on the critical part for determining the entailment value.

Kernels for capturing lexical-syntactic rules: Once place-holders are available in the entailment pairs, we can apply the model. This derives the maximal similarity between pairs of T and H based on the lexico-syntactic information encoded by the syntactic parse trees of T and H enriched with place-holders. More formally, the original kernel is based on the following equation:

$$maxSTK(\langle T, H \rangle, \langle T', H' \rangle) = \tag{2.7}$$

$$max_{c \in C}(STK(t(T, c), t(T', i)) + STK(t(H, c), t(H', i)),$$

where: *(i)* C is the set of all bijective mappings between the placeholders (*i.e.*, the possible variables) from $\langle T, H \rangle$ into $\langle T', H' \rangle$; *(ii)* $c \in C$ is a substitution function, which implements such mapping; *(iii)* $t(\cdot, c)$ returns the syntactic tree enriched with placeholders replaced by means of the substitution c; and *(iv)* $STK(\tau_1, \tau_2)$ is a tree kernel function.

The new semantic-syntactic kernel for lexical-syntactic rules, maxSSTK, increases the coverage of the matching between the pairs of texts and the pairs of hypotheses.

$$maxSSTK(\langle T, H \rangle, \langle T', H' \rangle) = \tag{2.8}$$

$$max_{c \in C}(SSTK(t(T, c), t(T', i)) + SSTK(t(H, c), t(H', i)),$$

2.6.5 Experiments

The aim of the experiments is to investigate if our RTE system exploiting syntactic semantic kernels (SSTK) can effectively derive generalized lexico-syntactic rules. In more detail, first, we determine the best lexical similarity suitable for the task, *i.e.* distributional vs. Wordnet-based approaches. Second, we derive qualitative and quantitative properties, which justify the selection of one with respect to the other.

For this purpose, we tested four different version of SSTK, *i.e.* using Path, WUP, BNC and Wiki lexical similarities on three different RTE datasets. These correspond to the three different challenges in which the development set was provided.

Experimental Setup:
We used the data from three recognizing textual entailment challenge: RTE-2, RTE-3, and RTE-5, along with the standard split between training and test sets. For these set of experiments, we did not use RTE-1 as it was differently built from the others and RTE-4 as it does not contain the development set.

We used the following publicly available tools: the Charniak Parser [Charniak, 2000] for parsing sentences and SVM-light-TK [Moschitti, 2006; Joachims, 1999b], in which we coded our new kernels for RTE. Additionally, we used the Jiang&Conrath (J&C) distance [Jiang & Conrath, 1997] computed with the `wn::similarity` package [Pedersen et al., 2004] to measure the similarity between T and H. This similarity is also used to define the text-hypothesis word overlap kernel (WOK).

The distributional semantics is captured by means of LSA: we used the java Latent Semantic Indexing (jLSI) tool [Giuliano, 2007]. In particular, we pre-computed the word-pair matrices for RTE-2, RTE-3, and

RTE-5. We build different LSA matrices from the British National Corpus (BNC) and Wikipedia (Wiki). The British National Corpus (BNC) is a balanced synchronic text corpus containing 100 million words with morpho-syntactic annotation. For Wikipedia, we created a model from the 200,000 most visited Wikipedia articles, after cleaning the unnecessary markup tags. Articles are our documents for creating the term-by-document matrix. Wikipedia provides the largest coverage knowledge resource developed by a community, besides the noticeable coverage of named entities. This further motivates the design of a similarity measure. We also consider two typical WordNet similarities (*i.e.*, Path and WUP, respectively) as described previously.

The main RTE model that we consider is constituted by three main kernels:

- WOK, *i.e.* the kernel based on only the text-hypothesis lexical overlapping features (this is an intra-pair similarity);
- STK, *i.e.* the sum of the standard tree kernel applied to the two text parse-trees and the two hypothesis parse trees;
- SSTK, *i.e.* the same as STK with the use of lexical similarities as explained previously;
- maxSTK and maxSSTK, *i.e.* the kernel for RTE, where the latter exploit similarity since it uses SSTK in Eq. 2.8.

Note that as our baseline, we considered the model presented in [Zanzotto et al., 2009], corresponds to the combination kernel: WOK+maxSTK. In addition to the role of lexical similarities we also study several combinations (we just need to sum the separated kernels), *i.e.* WOK+STK+maxSTK, SSTK+maxSSTK, WOK+SSTK+maxSSTK and WOK+maxSSTK.

		No Semantic	Wiki	BNC	Path	WUP
RTE-2	**j = 1**	63.12	63.5	62.75	62.88	**63.88**
	j = 0.9	63.38	**64.75**	62.26	63.88	64.25
RTE-3	**j = 1**	66.88	**67.25**	**67.25**	66.88	66.5
	j = 0.9	67.25	**67.75**	67.5	67.12	67.38
RTE-5	**j = 1**	65.5	**66.5**	65.83	66	66
	j = 0.9	65.5	**66.83**	65.67	66	66.33

Table 2.7: Accuracies of Plain (WOK+STK+maxSTK) Kernels and Semantic Lexico-Syntactic Rule (WOK+SSTK+maxSSTK) Kernels.

2.6.6 Results

Distributional vs. WordNet-based Semantics:
The first experiment compares the basic kernel, *i.e.* WOK+STK+maxSTK, with the new semantic kernel, *i.e.* WOK+SSTK+maxSSTK, where SSTK and maxSSTK encode four different kinds of similarities, BNC, Wiki, WUP and Path. The aim is twofold: understanding if semantic similarities can be effectively used to derive generalized lexico-syntactic rules and to determine the best similarity model.

Table 2.7 shows the results according to No Semantics, Wiki, BNC, Path and WUP. The three pairs of rows represent the results over the three different datasets, *i.e.*, RTE-2, RTE-3, and RTE-5. For each pair, we have two rows representing a different j parameter of SVM.[24] An increase of j augments the weight of positive with respect to negative examples and during learning it tunes-up the Recall/Precision rate. We use two values $j = 1$ (the default value) and $j = 0.9$ (selected during a preliminary experiment on a validation set on RTE-2). $j = 0.9$ was used to minimally increase

[24] j is a cost-factor by which training errors on positive examples outweight errors on negative examples (see [Morik et al., 1999]).

the Precision, considering that the semantic model tends to improve the Recall.

The results show that:

- Wiki semantics constantly improves the basic kernel (no Semantics) for any datasets or parameter.
- The distributional semantics is almost always better than the WordNet-based one.
- In one case WUP improves Wiki, *i.e.* 63.88 vs 63.5 and in another case BNC reaches Wiki, *i.e.* 67.25 but this happens for the default values of the j parameters, *i.e.* $j = 1$, which was not selected by our limited parameter validation.

Finally, the difference between the accuracies of the best Wiki kernels and the No Semantic kernels are statistically significant ($p < 0.05$).

Kernel Comparisons:
The previous experiments show that Wikipedia-based distributional semantics provides an effective similarity to generalize lexico-syntactic rules (features). As our RTE kernel is a composition of other basic kernels, we experimented with different combinations to understand the role of each component. Moreover, to obtain results independent of parametrization we used the default parameter j.

Table 2.8 reports the accuracy of different kernels and their combinations on different RTE datasets. Each row describes the results for each dataset and it is split in two according to the use of WOK or not in the RTE model. In the each column, the different kernels are reported. For example, the entry in the 4th column and the 2nd row refers to the accuracy of SSTK in combination with WOK, *i.e.* WOK+SSTK for the RTE-2. From the table we draw the following observations.

		STK	SSTK	maxSTK	maxSSTK	STK+maxSTK	SSTK+maxSSTK
RTE2	+WOK	61.5	61.12	63.88	**64.12**	63.12	**63.50**
	60.62	52.62	52.75	61.25	59.38	61.25	58.75
RTE3	+WOK	66.38	66.5	66.5	**67.0**	66.88	**67.25**
	66.75	53.25	54.5	62.25	64.38	63.12	63.62
RTE5	+WOK	62.0	62.0	64.83	64.83	65.5	*66.5*
	60.67	54.33	57.33	63.33	62.67	61.83	62.67

Table 2.8: Comparing different lexico-syntactic kernels with Wiki-based semantic kernels. Entries report accuracy percentages.

First, WOK produces a very high accuracy, *i.e.* 60.62, 66.75 and 60.67 and it is an essential component of RTE systems (as it was also observed by Kouylekov et al. [2011]) since its ablation always causes a large accuracy decrease. This is reasonable as the major source of information to establish entailment between sentences is their word overlap.

Second, STK and SSTK, when added to WOK, improve accuracy on RTE-2 and RTE-5 but not on RTE-3. This suggests the difficulty of exploiting syntactic information for RTE3.

Third, maxSTK+WOK relevantly improves WOK on RTE-2 and RTE-5 but fails in RTE-3. Again, the syntactic rules (with variables) which this kernel can provide are not enough general for RTE-3. In contrast, maxSSTK+WOK improves WOK on all datasets thanks to its generalization ability.

Finally, STK and SSTK added to maxSTK+WOK or to maxSSTK+WOK tend to produce an accuracy increase, although not in every condition.

Coverage and efficiency:
As already mentioned, the practical use of Wikipedia to design lexical sim-

ilarities is motivated by a large coverage. Moreover, Deriving similarities from other resources such as WordNet is more time-consuming. To prove our claim, we performed an analysis on the coverage and efficiency in computing the pair term similarity.

	BNC	WN	Wiki
RTE-2	0.55	0.42	0.83
RTE-3	0.54	0.41	0.83
RTE5-	0.45	0.34	0.82

Table 2.9: Coverage of the different resources for words of the three datasets.

Speed	Milliseconds
LSA	0.54
WN with POS	5.3
WN without POS	15.2

Table 2.10: The comparison in terms of speed calculated over 10000 pairs after loading the model.

Table 2.9 shows the coverage of the content words of the three datasets. The coverage of Wikipedia is about twice as large as that of the other resources in all experimented datasets.

Moreover, Table 2.10 shows that the computation of the similarity with the LSA matrix on Wikipedia is faster than using the WordNet similarity software [Pedersen et al., 2004]. Even if the accuracy of some WordNet models can reach the one based on Wikipedia, the latter is preferable for the smaller computational cost.

Comparison with previous works:
The results of our models show that lexical semantics for building more effective lexical-syntactic rules is promising. Here, we compare our approaches with other RTE systems to show that our results are indeed

state-of-the-art. Unfortunately, deriving a reasonable accuracy value to represent the state-of-the-art is extremely difficult as many factors can determine the final score. For example, the best systems in RTE-2 and RTE-3 [Giampiccolo et al., 2007] reported an accuracy 10% higher than other systems but also use resources that are not publicly available.

	Average Acc.	Our rank	# participants
RTE2	59.8	3rd	23
RTE3	64.5	4th	26
RTE5	61.5	4th	20

Table 2.11: Comparison with other approaches to RTE

Table 2.11 shows the average accuracy, the number of participants, and the rank of our system that we propose in this work. Our model accuracy is absolutely above the average and even ranks at the top. With respect to RTE-2 [Roy Bar-Haim et al., 2006], our system performs better than systems using semantic models based on FrameNet, indeed the best ranked system in this class scored only 62.5% [Burchardt et al., 2007]. Among systems using logical inference, our model ranks the 3rd out of 8 systems, and 2nd among systems using supervised machine learning models.

2.7 Summary

In this chapter, we reviewed the work related to the RTE problem. We started with the notion of textual entailment and introduced data resources available for this task. We then described different knowledge resources that have been used in the RTE scenario, including lexical databases and textual inference rules. In the same context, we introduced our contribution in providing more knowledge for RTE by using Wikipedia and parallel corpora and we proved that this lexical and phrase-based knowledge can help in improving performance [Mehdad et al., 2011; Kouylekov et al.,

2010a].

Furthermore, we described different approaches to RTE and compared them in different directions. We explained two novel methods to optimize edit distance based systems and algorithms using particle swarm optimization and genetic algorithm [Mehdad, 2009; Kouylekov & Negri, 2010], and reported experiments showing their significant improve on performance.

Finally, we proposed a novel syntactic-semantic tree kernel model for RTE [Mehdad et al., 2010a]. The comparative experiments across different RTE challenges and traditional systems show that our approach consistently and meaningfully achieve high accuracy, without requiring any adaptation or tuning.

Chapter 3

Cross-Lingual Textual Entailment

3.1 Introduction

Textual Entailment (TE) [Dagan & Glickman, 2004] has been proposed as a generic framework for modeling language variability. Given two texts T and H, the task is to decide if the meaning of H can be inferred from the meaning of T. So far, TE has been only applied in a *monolingual* setting, where both texts are assumed to be written in the same language. In this work, we propose and investigate a *cross-lingual* extension of TE, where we assume that T and H are written in different languages.

The great potential of integrating (monolingual) TE recognition components into NLP architectures has been reported in several works, such as question answering [Harabagiu & Hickl, 2006], information retrieval [Clinchant et al., 2006], information extraction [Romano et al., 2006], and document summarization [Lloret et al., 2008], discussed in Chapter 2.

To the best of our knowledge, mainly due to the absence of cross-lingual TE (CLTE) recognition components, similar integrations have not been achieved yet in any cross-lingual application. As a matter of fact, despite the great deal of attention that TE has received in recent years (also witnessed by five editions of the Recognizing Textual Entailment Challenge[1]),

[1] http://pascallin.ecs.soton.ac.uk/Challenges/RTE/

interest for cross-lingual extensions has not been mainstream research for TE, which till date the main focus was only on the English language.

Nevertheless, the strong interest towards cross-lingual NLP applications (both from the market and research perspectives, as demonstrated by successful evaluation campaigns such as CLEF[2]) is, to our view, a good reason to start investigating CLTE. Along with such direction, research can now benefit from recent advances in other fields, especially machine translation (MT), and the availability of: *i)* large amounts of parallel and comparable corpora in many languages, *ii)* open source software to compute word-alignments from parallel corpora, and *iii)* open source software to set-up strong MT baseline systems. We strongly believe that all these resources can potentially help in developing inference mechanisms on multilingual data.

Building on these considerations, this chapter aims to put the cross-lingual Textual Entailment task as the main research problem of this thesis, in order to allow for semantic inference across languages in different NLP applications. With the awareness that MT approaches can play an important role in moving toward this direction, we also devote particular attention to exploit MT techniques in approaching the problem of recognizing textual entailment across languages. Among these, we also adopt CLTE to support real world NLP applications and tasks such as: *i)* automatic alignment of text portions that express the same meaning in different languages (Chapter 4), and *ii)* automatic evaluating the adequacy of machine translation output without usding reference translations (Chapter 5).

[2]www.clef-campaign.org/

3.2 Cross Lingual Textual Entailment

This section defines CLTE, highlighting some issues and proposing possible approaches to the problem. We also mention the lexical and knowledge resources which are potentially useful in our approach.

3.2.1 Definition

Adapting the definition of TE we define CLTE as a relation between two natural language portions in different languages, namely a text T (*e.g.* in English), and a hypothesis H (*e.g.* in Spanish), that holds if a human after reading T would infer that H is most likely true, or otherwise stated, the meaning of H can be entailed (inferred) from T .

In other words, in developing the idea of CLTE, we should be able to predict whether there is an entailment at the multi-lingual level over portions of texts in different languages. Example 1 shows two portion of texts in English and Spanish, where the entailment relation holds.

Example 1.
***T**: Wolfgang Amadeus Mozart was born in Salzburg, capital of the sovereign Archbishopric of Salzburg, in what is now Austria.*
***H**: Mozart nació en Austria.*
***Entailment**: YES*

The task of CLTE is inherently difficult, as it adds issues related to the multilingual dimension to the complexity of semantic inference at the textual level. For instance, the reliance of current monolingual TE systems on lexical resources (*e.g.* WordNet, VerbOcean, FrameNet) and deep processing components (*e.g.* syntactic and semantic parsers, co-reference resolution tools, temporal expressions recognizers and normalizers) has to confront, at the cross-lingual level, with the limited availability of lexi-

cal/semantic resources covering multiple languages, the limited coverage of the existing ones, and the burden of integrating language-specific components into the same cross-lingual architecture.

3.2.2 Approaches

In order to approach the CLTE problem, we can see two main orthogonal directions: *i)* simply bring CLTE back to the monolingual case by translating H into the language of T or vice-versa; *ii)* try to develop and integrate cross-lingual techniques inside the TE recognition process. In the following, we briefly overview and motivate each approach.

Basic Approaches

The overgrowing amount of parallel data, as well as the incremental efforts on MT research, motivates to import the current available technology in MT into CLTE, as an initial approach aiming to recognize textual entailment and semantic inference across languages. In this way, the simplest approach is to add a MT component to the front-end of an existing TE engine. In this method, assuming that T is in English and H in another language, or both in different languages than English, taking advantage of a MT system, we only require to translate the hypotheses or both to English, then accordingly, approach the problem in a monolingual fashion.

For instance, let the Spanish hypothesis H (*e.g.* in Example 1) be translated into English and then run the TE engine on T and the translation of H. In this way, regardless of the entailment engine, we only need to have a translation system. Figure 3.1 shows a sketch view of the basic approach framework (left figure).

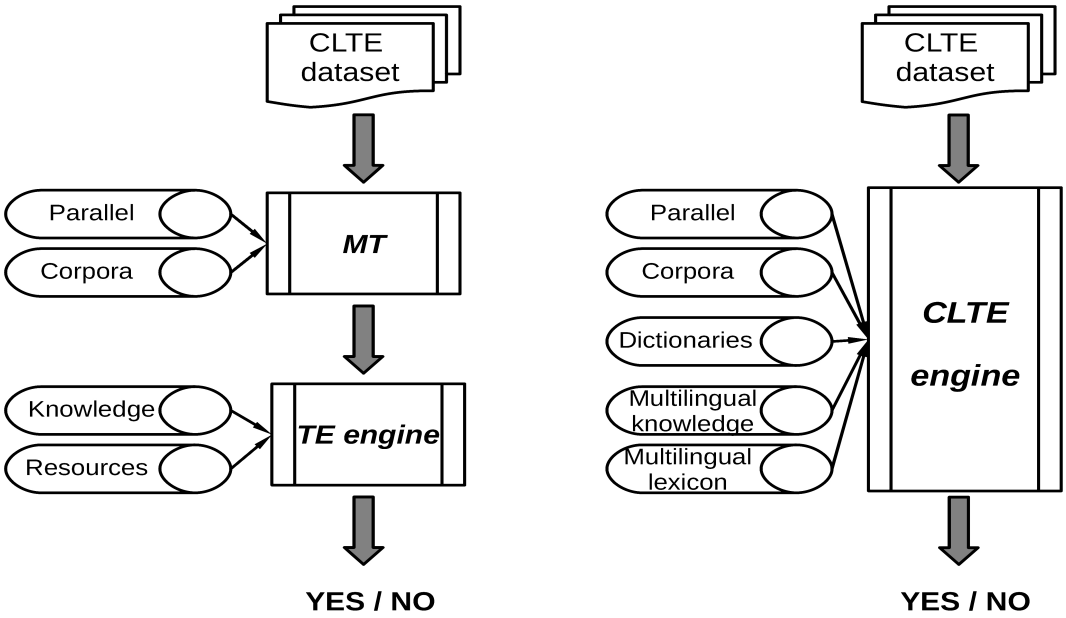

Figure 3.1: Left: basic approach by adding a MT component to the front-end of an existing TE engine. Right: advanced approach by tighter integration of MT and TE algorithms and techniques.

Advanced Approaches

The purpose of advanced methods is to move towards a cross-lingual TE approach that takes advantage of a tighter integration of MT and TE algorithms and techniques. This could result in methods for recognizing TE across languages without translating the texts and, in principle, with a lower complexity. When dealing with phrase-based statistical MT, a possible approach is to extract information from translation phrase tables, as a source of knowledge, to enrich the inference and entailment rules which could be used in any entailment system.

As an example the entailment relations between the French phrase "*ordinateur portable*" and the English phrase *laptop*", or between the German phrase "Europaeischen Union" and the English word "*Europe*" could be captured from parallel corpora through statistical phrase-based MT approaches. In another word, focusing on Example 1, applying these methods

we can detect the entailment between:

- "*Wolfgang Amadeus Mozart*" → "*Mozart*"
- "*Salzburg, in what is now Austria*" → "*Autriche*"

In this way, it would help to recognize the entailment without translating the hypothesis to English. There are several implications that make this approach interesting. First of all, the acquired rules could as well enrich the available multilingual resources and dictionaries such as MultiWordNet,[3] which will be more explained in the next section. In addition, such approaches can employ inference mechanisms and semantic knowledge sources to augment existing MT methods, leading to improvements in the translation quality. Figure 3.1 shows a sketch view of the advanced approach framework (right figure).

Cross-lingual Knowledge Resources

Despite the consensus on the usefulness of lexical knowledge for textual inference, determining the actual impact of these resources is not straightforward, as they always represent one factor in complex architectures that use them in different ways. As emerges from the ablation tests reported in Bentivogli et al. [2010b], even the most common resources have a positive impact on some systems and a negative impact on others (as discussed in Chapter 2). Some previous works [Bannard & Callison-Burch, 2005; Zhao et al., 2009; Kouylekov et al., 2010a] indicate, as main limitations of the mentioned resources, the limited coverage, the low precision, and the fact that they are mostly suitable to capture relations between single words.

Addressing CLTE we have to face additional and more problematic issues related to: *i)* the stronger need of lexical knowledge, and *ii)* the limited

[3] http://multiwordnet.fbk.eu/

availability of multilingual lexical resources. As regards the first issue, it's worth noting that in the monolingual scenario simple "bag of words" (or "bag of n-grams") approaches are *per se* sufficient to achieve results above the baseline[4]. In contrast, their application in the cross-lingual setting is not a viable solution due to the impossibility to perform direct lexical matches between texts and hypotheses in different languages. This situation makes the availability of multilingual lexical knowledge a necessary condition to bridge the language gap.

However, with the exceptions represented by WordNet and Wikipedia, most of the aforementioned resources are available only for English. Multilingual lexical databases aligned with the English WordNet (*e.g.* MultiWordNet [Bentivogli et al., 2002]) have been created for several languages, with different degrees of coverage. As an example, the 57,424 synsets of the Spanish section of MultiWordNet aligned to English cover just around 50% of the WordNet's synsets, thus making the coverage issue even more problematic than for TE. As regards Wikipedia, the cross-lingual links between pages in different languages offer a possibility to extract lexical knowledge useful for CLTE. However, due to their relatively small number (especially for some languages), bilingual lexicons extracted from Wikipedia are still inadequate to provide acceptable coverage. In addition, featuring a bias towards named entities, the information acquired through cross-lingual links can at most complement the lexical knowledge extracted from other resources (*e.g.* bilingual dictionaries).

[4]Within the framework of the RTE challenge, a naive baseline of 50% could be estimated by simply labeling all entailments as true (or as false). We also proposed another baseline by measuring the similarity estimated as the degree of word overlap between T and H [Mehdad & Magnini, 2009c].

3.3 Basic Solution (Pivoting)

As a first step to approach CLTE, we propose a "*basic solution*", that brings CLTE back to the monolingual scenario by translating H into the language of T. Despite the advantages in terms of modularity and portability of the architecture, and the benefit of exploiting monolingual knowledge resources, this approach suffers from one main limitation which motivates the investigation on alternative solutions. Decoupling machine translation and TE, in fact, ties CLTE performance to the availability of MT components, and to the quality of the translations. As a consequence, on one side translation errors propagate to the TE engine hampering the entailment decision process. On the other side such unpredictable errors reduce the possibility to control the behaviour of the engine, and devise *ad-hoc* solutions to specific entailment problems.

The main purposes of our experiments with basic solution is two-fold. First, to verify the feasibility of CLTE and proving that this task, to some extent, can be approached even with a basic solution, in the absence of cross-lingual components. Second, to estimate the affect of noise introduced by an automatic translation as well as setting baseline results to be further improved, using the advanced solution.

3.3.1 Experiment 1: Feasibility Study

In order to create a realistic and standard setting, we took advantage of the available RTE data, selecting the RTE-3 development set and manually translating the hypotheses into French. Since the manual translation requires trained translators, and due to time and logistics constraints, we obtained 520 translated hypotheses (randomly selected from the entire RTE-3 development set) which built our bilingual entailment corpus for evaluation.

Our decisions build on several motivations. First of all, the reason for setting English and French as a first language pair for experiments is to rely on higher quality translation models, and larger amounts of parallel data for future improvements. Second, the reason for translating the hypotheses is that, according to the notion of TE, they are usually shorter, less detailed, and barely complex in terms of syntax and concepts with respect to the texts. This makes them easier to translate preserving the original meaning. Finally, from an application-oriented perspective, working with English Ts seems more promising due the richness of English data available (*e.g.* in terms of language variability, and more detailed elaboration of concepts). This increases the probability to discover entailment relations with Hs in other languages.

In the initial step, following our basic approach, we translated the French hypotheses to English using Google[5] and Moses.[6] We trained a phrase-base translation model using Europarl[7] and News Commentary parallel corpora in Moses, applying a 6-gram language model trained on the New York Times portion of the English Gigaword corpus.[8] More details will be provided in the next sections.

As a TE engine , we used the EDITS package (Edit Distance Textual Entailment Suite),[9] as an open source software package based on edit distance algorithms, which computes the T-H distance as the cost of the edit operations (*i.e.* insertion, deletion and substitution) that are necessary to transform T into H. By defining the edit distance algorithm and a cost scheme (*i.e.* which defines the costs of each edit operation), this package is able to learn a distance model over a set of training pairs, which is used

[5]http://translate.google.com

[6]Moses is a statistical machine translation system that allows to automatically train translation models for any language pair. This package is available at http://www.statmt.org/moses/

[7]http://www.statmt.org/europarl/

[8]http://www.ldc.upenn.edu

[9]http://edits.fbk.eu/

	Orig.	**Google**	**Moses** 1st best	**Moses** 30 best	**Moses** > 0.4
Accuracy	63.48	63.48	61.37	62.90	62.90

Table 3.1: Feasibility study: accuracy results comparison over 520 test pairs English-French dataset.

to decide if an entailment relation holds over each test pair.[10]

In order to obtain a monolingual TE model, we trained and optimize our model [Mehdad & Magnini, 2009a] on the RTE-3 test set, to reduce the over-fitting bias, since our original data was created over the RTE-3 development set. Moreover, we used a set of lexical entailment rules extracted from Wikipedia and WordNet, as described in Mehdad et al. [2009b]. To begin with, we used this model to classify the created cross-lingual entailment corpus in three different settings: *i)* hypotheses translated by Google, *ii)* hypotheses translated by Moses (1^{st} best), and *iii)* the original RTE-3 monolingual English pairs.

Results reported in Table 3.1 show that using Google as a translator, in comparison with the original manually-created data, does not cause any drop in performance. This confirms that merely translating the hypothesis using a good translation model (Google) is a feasible and promising direction for CLTE. Knowing that Google has one of the best French-English translation models, the downtrend of results using Moses translator, in contrast with Google, is not out of our expectation. This result also set the Google translate as a strong MT system for the rest of our experiments in this chapter.

Trying to bridge this gap brings us to the next round of experiments, where we extracted the n-best translations produced by Moses, to have a richer lexical variability, beneficial for improving the TE recognition. The graph in Figure 3.3 shows an incremental improvement when the n-

[10]More details about the models and system has been explained in Chapter 2.

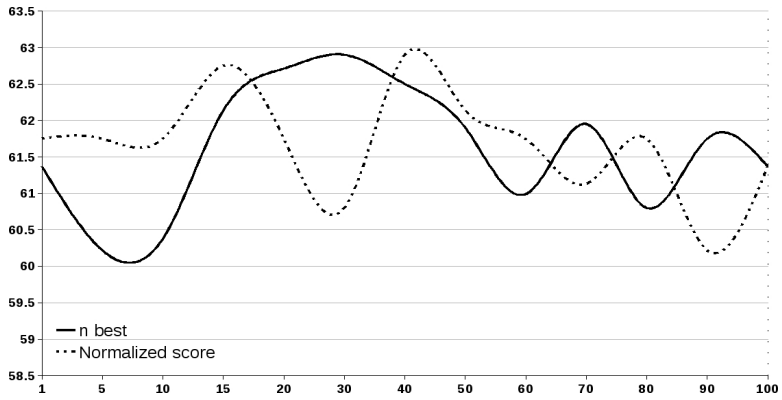

Figure 3.2: Accuracy gained by n-best Moses translations shows an incremental improvement when the n-best translated hypotheses are used.

best translated hypotheses are used. Besides that, trying to reach a more monotonic distribution of the results, we normalized the ranking score (from 0 to 1) given by Moses, and in each step we chose the first n results over a normalized score. In this way, having the hypotheses with the score of above 0.4, we achieved the highest accuracy of 62.9%. This is exactly equal to adopting the 30-best hypotheses translated by Moses. Using this method, we could improve the performance up to 1.5% above the 1^{st} best results, achieving almost the same level of performance obtained with Google. These results also prove that TE can be used to estimate the quality of translations, and motivates another interesting application of CLTE that will be discussed in Chapter 5. Overall, the feasibility study presented a preliminary investigation towards Cross-lingual Textual Entailment, proving the viability of moving this direction.

3.3.2 Experiment 2: Verification

Using the basic solution (pivoting) and a different dataset, in this section, we conduct various experiments taking advantage of different knowledge resources to verify if in the cross-lingual scenario, we can achieve a result comparable to those obtained in the monolingual TE. Moreover, we try to

measure the effectiveness of various monolingual knowledge sources for the basic solution.

Dataset

In order to confront our result with the monolingual RTE results, we experiment with the original RTE-3 and the RTE3-derived CLTE dataset. The CLTE dataset used for our experiments is an English-Spanish entailment corpus obtained from the original RTE-3 dataset by translating the English hypothesis into Spanish. It consists of 1600 pairs derived from the RTE-3 development and test sets (800+800). Translations have been crowdsourced, using the CrowdFlower[11] channel to Amazon Mechanical Turk[12] (MTurk), adopting the methodology which is elaborated in Chapter 6. The method relies on translation-validation cycles, defined as separate jobs routed to MTurk's workforce. Translation jobs return one Spanish version for each hypothesis. Validation jobs ask multiple workers to check the correctness of each translation using the original English sentence as reference. At each cycle, the translated hypothesis accepted by the majority of trustful validators[13] are stored in the CLTE corpus, while wrong translations are sent back to workers in a new translation job. Although the quality of the results is enhanced by the possibility to automatically weed out untrusted workers using gold units, we performed a manual quality check on a subset of the acquired CLTE corpus. The validation, carried out by a Spanish native speaker on 100 randomly selected pairs after two translation-validation cycles, showed the good quality of the collected material, with only 3 minor "errors" consisting in controversial but substantially acceptable translations reflecting regional Spanish variations. To

[11] http://crowdflower.com/

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

[13] Workers' trustworthiness can be automatically determined by means of hidden gold units randomly inserted into jobs.

conduct our experiments based on the basic solution, we then translated the Spanish hypotheses of the dataset into English using Google Translate. The T-H pairs in both datasets were annotated using the TreeTagger [Schmid, 1995] and the Snowball stemmer [Porter, 2001] with token, lemma, and stem information.

Algorithm

In order to maximize the usage of lexical knowledge, our entailment decision criterion is based on similarity scores calculated with a phrase-to-phrase matching process. phrase in our approach is an *n-gram* composed of one or more (up to 5) consecutive words, excluding punctuation. Entailment decisions are assigned combining phrasal matching scores ($Score_n$) calculated for each level of *n-grams* (*i.e.* considering the number of *1-grams*, *2-grams*,..., *5-grams* extracted from H that match with *n-grams* in T). Phrasal matches, performed either at the level of tokens, lemmas, or stems, can be of two types:

1. **Exact**: in the case that two phrases are identical at one of the three levels (token, lemma, stem).

2. **Lexical**: in the case that two different phrases can be mapped through entries of the resources used to bridge T and H (*i.e.* phrase tables, paraphrases tables, dictionaries or any other source of lexical knowledge).

For each phrase in H, we first search for exact matches at the level of token with phrases in T. If no match is found at a token level, the other levels (lemma and stem) are attempted. Then, in case of failure with exact matching, lexical matching is performed at the same three levels. To reduce redundant matches, the lexical matches between pairs of phrases which have already been identified as exact matches are not considered.

Input: T and H pair represented at the level of token, lemma and stem
Output: Matching score at each n-gram level n
$T = ngrams(T)$;
$H = ngrams(H)$;
foreach *n=1 to 5* **do**
  $Match_n = 0$;
  foreach *type=exact,lexical* **do**
    foreach $h \in H(n)$ **do**
      foreach $form$*=token,stem,lemma* **do**
        if *PhraseMatch*$(h_{form}, T$*,type)* **then**
          $Match_n = Match_n + 1$;
          next h;
        end
      end
    end
  end
  $Match_n = \frac{Match_n}{|H(n)|}$;
end

Algorithm 1: Phrase matching algorithm

Once the matching phase for each *n-gram* level has been concluded, the number of matches $Match_n$ and the number of phrases in the hypothesis $H(n)$ is used to estimate the portion of phrases in H that are matched at each level n. The phrasal matching score for each *n-gram* level is described in Algorithm 1. Since languages can express the same meaning with different amounts of words, a phrase with length n in H (*i.e* h in Algorithm 1) can match a phrase with any length in T (*i.e* T in Algorithm 1).

To combine the phrasal matching scores obtained at each *n-gram* level, and optimize their relative weights, we trained a Support Vector Machine classifier, SVMlight [Joachims, 1999a], using each score as a feature. Our main motivations in using SVM are summarized as follows.

- SVMs have been successfully exploited in a number of NLP tasks and

achieved state-of-the-art performance among other algorithms.

- The generalization capability of SVM is not depending on the feature vector dimension.
- Feature combination in SVM is more efficient in terms of computational complexity, thus adding more features does not increase the computational cost dramatically.
- Increasing the input dimension, does not increase the number of optimizing parameters.

Knowledge sources

In order to compare the results between the monolingual and cross lingual datasets, we used different monolingual knowledge sources as explained in Chapter 2, namely:

1. Paraphrase table (PPT): we used a publicly available[14] paraphrase database for English. Moreover, in order to experiment with different paraphrase sets providing different degrees of coverage and precision, we pruned the main paraphrase table based on the probabilities, associated to its entries, of 0.1, 0.2 and 0.3. The number of phrase pairs extracted varies from 6 million to about 80,000, with an average of 3.2 words per phrase.
2. WordNet (WN): WordNet 3.0 has been used to extract a set of 5,396 pairs of words connected by the hyponymy and synonymy relations.
3. VerbOcean (VO): VerbOcean has been used to extract 18,232 pairs of verbs in the same way discussed in Chepter 3 Section 2.3.3.

[14] http://www.cs.cmu.edu/alavie/METEOR

Dataset	WN	VO	WIKI	PPHT	PPHT 0.1	PPHT 0.2	PPHT 0.3	AVG
RTE3	61.88	62.00	61.75	62.88	**63.38**	**63.50**	63.00	62.37
RTE3-derived	62.62	61.5	60.5	62.88	**63.50**	62.00	61.5	-

Table 3.2: Accuracy results on monolingual setting (pivoting) using different lexical resources.

4. Wikipedia (WP): we performed Latent Semantic Analysis (LSA) over Wikipedia using the jLSI tool [Giuliano, 2007] to measure the relatedness between words in the dataset. Then, we filtered all the pairs with similarity lower than 0.7 as proposed by Kouylekov et al. [2010a]. In this way we obtained 13,760 word pairs.

Results

The comparison with the results achieved on original monolingual data (RTE-3) and the one obtained by automatically translating the Spanish hypotheses (RTE3-derived row in Table 3.2) leads to three main observations.

1. We notice that dealing with MT-derived inputs, the optimal pruning threshold changes from 0.2 to 0.1, leading to the highest result of 63.50% Accuracy. This suggests that the noise introduced by incorrect translations can be partially tackled by increasing the coverage of the paraphrase table.

2. In line with the purpose of our experiments, the results obtained over the MT-derived corpus are equal to those we achieve over the original RTE-3 dataset (*i.e.* 63.50%). This further proves that using a suitable algorithm with a high coverage source of knowledge, we can achieve results comparable to those obtained in monolingual TE.

3. As regards the other resources used for comparison, the results

achieved with PPT always outperform the results obtained using VO, WP and WN. This can be explained by the high coverage of PPT, and the possibility of matching longer phrases in H preserving more contextual information.

In light of this, we suggest that the lexical knowledge extracted from parallel data (PPT) can be successfully used to approach the CLTE task, with the basic solution. To answer the main question of this section, besides measuring the effectiveness of different knowledge sources in dealing with the CLTE pivoting approach, we obtain the comparable results with RTE-3 monolingual scenario and we outperform the average results obtained by the participant of RTE-3 campaign.

3.4 Advanced Solution (cross-lingual)

As a first step to approach CLTE, in the last section, we proposed a "basic solution", that brings CLTE back to the monolingual scenario by translating H into the language of T. Despite the advantages in terms of modularity and portability of the architecture, and the promising experimental results, this approach suffers from one main limitation which motivates the investigation on alternative solutions. Decoupling Machine Translation (MT) and TE, in fact, ties CLTE performance to the availability of MT components, and to the quality of the translations. As a consequence, on one side translation errors propagate to the TE engine hampering the entailment decision process. On the other side such unpredictable errors reduce the possibility to control the behaviour of the engine, and devise *ad-hoc* solutions to specific entailment problems.

This section investigates the idea of a tighter integration and joint optimization of MT and TE algorithms and techniques. Our aim is to embed and integrate cross-lingual techniques inside the TE recognition process in

order to avoid any dependency on external MT components, and eventually gain full control of the system's behaviour. Along this direction, we start from the acquisition and use of lexical knowledge, which represents the basic building block of any TE system. Our experiment with different sources of multilingual lexical knowledge aims at addressing the following questions:

1. What is the potential of the existing multilingual lexical resources to approach CLTE? To answer this question we experiment with lexical knowledge extracted from bilingual dictionaries, and from a multilingual lexical database. Such experiments show two main limitations of these resources, namely: *i)* their limited coverage, and *ii)* the difficulty to capture contextual information when only associations between single words (or at most named entities and multiword expressions) are used to support inference.

2. Does MT provide useful resources or techniques to overcome the limitations of the existing resources? We envisage several directions in which inputs from MT research may enable or improve CLTE. As regards the resources, phrase and paraphrase tables extracted from bilingual parallel corpora can be exploited as an effective way to capture both lexical relations between single words, and contextual information useful for inference. As regards the algorithms, statistical models based on co-occurrence observations, similar to those used in MT to estimate translation probabilities, may contribute to estimate entailment probabilities in CLTE.

3. Can we take advantage of relevant semantic and syntactic information in cross-lingual scenario? By integrating linguistically motivated syntactic and semantic features, we propose another novel approach

that uses a rich set of features to improve over the lexical based CLTE results.

The remainder of this section tries to address the questions above and showing the results of our experiments, concluding the effectiveness of our cross-lingual approach for CLTE.

3.4.1 Exploiting Parallel Corpora for CLTE

The limitations of bilingual lexical resources, in terms of coverage and availability, has always been an issue for cross-lingual applications. Bilingual parallel corpora represent a possible solution to overcome the inadequacy of the existing resources, and to implement a portable approach for CLTE. To this aim, we exploit parallel data to: *i)* learn alignment criteria between phrasal elements in different languages, *ii)* use them to automatically extract lexical knowledge in the form of *phrase tables*, and *iii)* use the obtained phrase tables to create monolingual *paraphrase tables* (as it was explained in Chapter 2 and previous section).

Given a cross-lingual T/H pair (with the text in l_1 and the hypothesis in l_2), our approach leverages the vast amount of lexical knowledge provided by phrase and paraphrase tables to map H into T. We perform such mapping with two different methods. The **first method** uses a single phrase table to directly map phrases extracted from the hypothesis to phrases in the text. In order to improve our system's generalization capabilities and increase the coverage, the **second method** combines the phrase table with two monolingual paraphrase tables (one in l_1, and one in l_2). This allows to:

1. use the paraphrase table in l_2 to find paraphrases of phrases extracted from H;

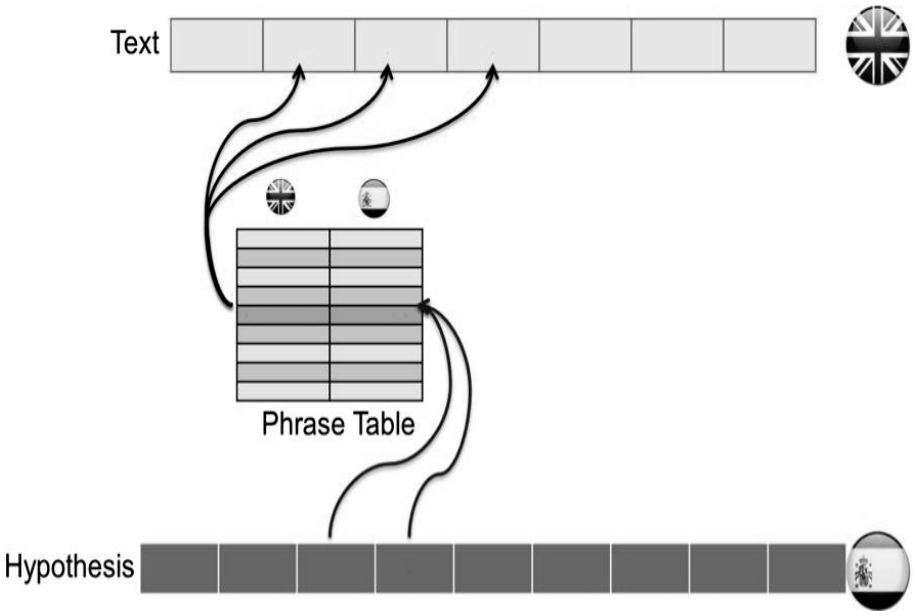

Figure 3.3: Using a phrase table for CLTE.

2. map them to entries in the phrase table, and extract their equivalents in l_1;

3. use the paraphrase table in l_1 to find paraphrases of the extracted fragments in l_1;

4. map such paraphrases to phrases in T.

With the second method, phrasal matches between the text and the hypothesis are indirectly performed through paraphrases of the phrase table entries. Figures 3.3 and 3.4 demonstrate both methods in using phrase and paraphrase tables.

The final entailment decision for a T/H pair is assigned considering a model learned from the similarity scores based on the identified phrasal matches. In particular, "YES" and "NO" judgements are assigned considering the proportion of words in the hypothesis that are found also in the text. This way to approximate entailment reflects the intuition that, as a directional relation between the text and the hypothesis, the full content

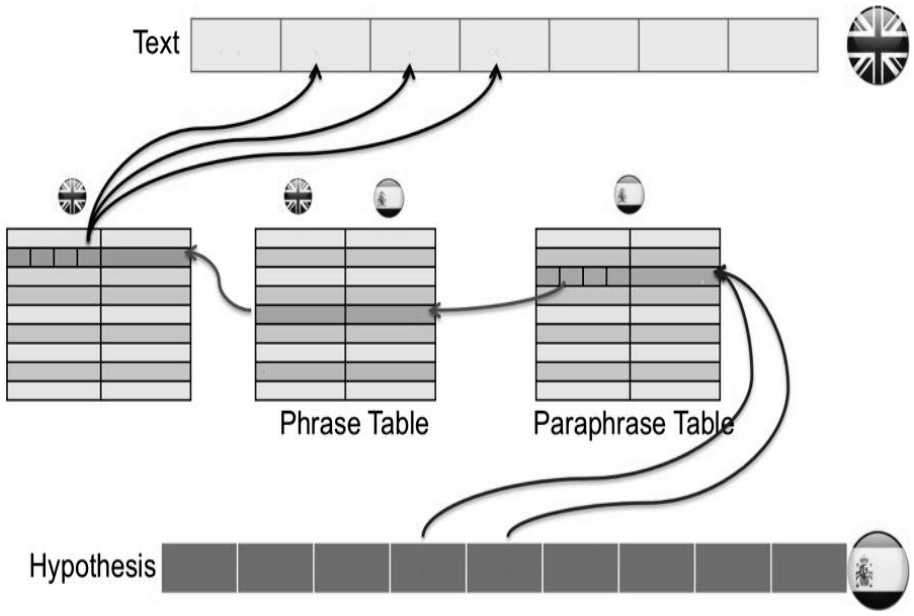

Figure 3.4: Combining phrase and paraphrase tables for CLTE.

of H has to be found in T.

Extracting English-Spanish Phrase and Paraphrase Tables

Phrase tables (PT) contain pairs of corresponding phrases in two languages, together with association probabilities. They are widely used in statistical machine translation as a way to figure out how to translate input in one language into output in another language [Koehn et al., 2003]. There are several methods to build phrase tables. The one adopted in this work consists in learning phrase alignments from a word-aligned bilingual corpus. In order to build English-Spanish phrase tables for our experiments, we used the freely available Europarl V.4, News Commentary and United Nations Spanish-English parallel corpora released for the WMT10 Shared Translation Task.[15] We run the TreeTagger for tokenization, and used the Giza++ [Och & Ney, 2000] toolkit to align the tokenized corpora at the word level. Subsequently, we extracted the bi-lingual phrase table

[15]http://www.statmt.org/wmt10/

from the aligned corpora using the Moses toolkit [Koehn et al., 2007]. Since the resulting phrase table was very large, we pruned all the entries with identical content in the two languages, and the ones containing phrases longer than 5 words in one of the two sides. In addition, in order to experiment with different phrase tables providing different degrees of coverage and precision, we extracted 7 phrase tables from the pruned one based on the direct phrase translation probabilities of $0.01, 0.05, 0.1, 0.2, 0.3, 0.4$ and 0.5. The resulting phrase tables range from 76 to 48 million entries, with an average of 3.9 words per phrase.

Paraphrase tables (PPT) contain pairs of corresponding phrases in the same language, possibly associated with probabilities. They proved to be useful in a number of NLP applications such as natural language generation [Iordanskaja et al., 1991], multidocument summarization [McKeown et al., 2002], automatic evaluation of machine translation [Denkowski & Lavie, 2010], and textual entailment [Dinu & Wang, 2009].

One of the proposed methods to extract paraphrases relies on a pivot-based approach using phrase alignments in a bilingual parallel corpus [Bannard & Callison-Burch, 2005]. With this method, all the different phrases in one language that are aligned with the same phrase in the other language are extracted as paraphrases. After the extraction, pruning techniques [Snover et al., 2009] can be applied to increase the precision of the extracted paraphrases.

In our work we used available paraphrase databases for English and Spanish[16] which have been extracted using the method previously outlined. We used the same method discussed in Section 3.3.2 to extract different sets of paraphrases.

[16] http://www.cs.cmu.edu/ alavie/METEOR

Experiments

The dataset used for our experiments is the RTE3-derived English-Spanish entailment corpus which was used in our previous experiments. The T-H pairs in the collected English-Spanish entailment corpus were annotated using the TreeTagger and the Snowball stemmer[17] with token, lemma, and stem information.

We use the PT and PPT as lexical knowledge to calculate a matching score (see Algorithm 1), as the number of n-grams in H that match with phrases in T divided by the number of n-grams in H. Using each score as a feature, we used SVMlight [Joachims, 1999a] to combine and weight features at different levels of ngrams. For comparison with the extracted phrase and paraphrase tables, we use a large bilingual dictionary and MultiWordNet as alternative sources of lexical knowledge.

1. Bilingual dictionaries (DIC) allow for precise mappings between words in H and T. To create a large bilingual English-Spanish dictionary we processed and combined the following dictionaries and bilingual resources:

   - Universal dictionary database[18]: 9,944 enteries.
   - Wiktionary database[19]: 5,866 enteries.
   - Omegawiki database[20]: 8,237 enteries.
   - Wikipedia interlanguage links[21]: 7,425 enteries.

   The resulting dictionary features 53,958 unique entries, with an average length of 1.2 words.

[17] http://snowball.tartarus.org/
[18] http://www.dicts.info/
[19] http://en.wiktionary.org/
[20] http://www.omegawiki.org/
[21] http://www.wikipedia.org/

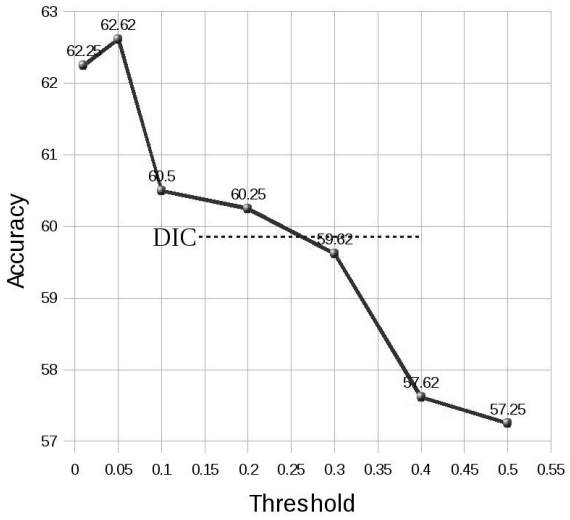

Figure 3.5: Accuracy on CLTE using phrase tables with different pruning thresholds.

2. MultiWordNet (MWN) allows to extract mappings between English and Spanish words connected by entailment-preserving semantic relations. The extraction process is dataset-dependent, as it checks for synonymy and hyponymy relations only between terms found in the dataset. The resulting collection of cross-lingual words associations contains 36794 pairs of lemmas.

Results

This section reports the percentage of correct entailment assignments (accuracy), contrasting the different sources of lexical knowledge.

Initially, in order to find a reasonable trade-off between precision and coverage, we used the 7 phrase tables extracted considering different pruning thresholds. Figure 3.5 shows that with the pruning threshold set to 0.05, we obtain the highest result of 62.62% on the test set. The curve demonstrates that, although with higher pruning thresholds we retain more precise phrase pairs, their smaller number provides limited coverage leading to lower results. In contrast, the large coverage obtained with the

MWN	DIC	PHT	PPHT	Acc.	δ
x				55.00	0.00
	x			59.88	+4.88
		x		**62.62**	+7.62
		x	x	**62.88**	+7.88

Table 3.3: Accuracy results on CLTE using different lexical resources.

pruning threshold set to 0.01 leads to a slight performance decrease due to less precise phrase pairs.

Once the threshold has been set, in order to prove the effectiveness of information extracted from bilingual corpora, we conducted a series of experiments using the different resources.

As it can be observed in Table 3.3, the highest results are achieved using the phrase table, both alone and in combination with paraphrase tables (62.62% and 62.88% respectively). These results suggest that, with appropriate pruning thresholds, the large number and the longer entries contained in the phrase and paraphrase tables represent an effective way to:

1. Obtain high coverage.

2. Capture cross-lingual associations between multiple lexical elements.

This allows to overcome the bias towards single words featured by dictionaries and lexical databases.

As regards the other resources used for comparison, the results show that dictionaries substantially outperform MWN. This can be explained by the low coverage of MWN, which entries also represent weaker semantic relations (preserving entailment, but with a lower probability to be applied) than the direct translations between terms contained in the dictionary.

Overall, our results suggest that the lexical knowledge extracted from parallel data can be successfully used to approach the CLTE task.

3.4.2 Beyond Lexical Features

The above mentioned advanced solution to the CLTE problem is based on the assumption that parallel data represent an ideal source of lexical knowledge to cross the language barrier between texts and hypotheses. Building on this assumption, CLTE has been modeled as a phrase matching problem that takes advantage of dictionaries and phrase tables extracted from bilingual parallel corpora to determine the number of word sequences (at the level of tokens, lemmas, or stems) in the hypothesis that can be mapped to word sequences in the text. According to this solution, a *semantic* judgement about entailment is made exclusively on the basis of *lexical* evidence. Although quite effective in cross-lingual datasets derived from the RTE-like setting, such approximation falls short of providing a reliable method for more complex scenarios, like the one addressed here. On the one side, in the traditional RTE-derived datasets only unidirectional entailment relations from T to H have to be determined, and the full mapping of the hypothesis into the text usually provides enough evidence for a positive entailment judgement. On the other side, textual entailment, in nature, deals with multi-directional entailment checking, where the correlation between the proportion of matching terms and the correct entailment decisions is less strong. In such framework, for instance, the full mapping of the hypothesis into the text is *per se* not sufficient to discriminate between forward entailment and semantic equivalence.

To cope with these issues, we explore the potential contribution of syntactic and semantic features, as a complement to lexical ones in a supervised learning framework. In order to enrich the feature space beyond pure lexical match through phrase table entries, our model builds on two additional feature sets, respectively derived from: *i)* dependency relations, and *ii)* semantic phrase tables.

T	**H**
Mozart was born in Salzburg	Mozart nació en 1756.
1. born/VERB — **Subj** — Mozart/NOUN 2. born/VERB — Spec — was/VERB 3. born/VERB — **Prep** — in/PRP 4. in/PRP — Pobj — Salzburg/NOUN	1. nació/VERB — **Subj** — Mozart/NOUN 2. nació/VERB — **Prep** — en/PRP 3. en/PRP — Pobj — 1756/CARD
DR matching (*DR_match*)	
Subj $= 1/1$, **Prep** $= 1/1$, *Pobj* $= 0/1$	
1. born/VERB — **Subj** — Mozart/NOUN # nació/VERB — **Subj** — Mozart/NOUN 2. born/VERB — **Prep** — in/PRP # nació/VERB — **Prep** — en/PRP	

Table 3.4: Dependency Relation (DR) matching between an English text and a Spanish hypothesis.

Dependency Relation Matching

Dependency Relation (DR) matching targets the increase of CLTE precision. By adding syntactic constraints to the matching process, DR features aim to reduce wrong matches often occurring at the lexical level. For instance, the contradiction between "*Yahoo acquired Overture*" and "*Overture compró Yahoo*" is evident when syntax (in this case subject-object inversion) is taken into account, but can not be caught by bag-of-words methods.

We define a dependency relation as a triple that connects pairs of words through a grammatical relation. For example, *"nsubj (loves, John)"* is a dependency relation with head *loves* and dependent *John* connected by the relation *nsubj*, which means that *"John"* is the *subject* of *"loves"*. DR matching captures similarities between dependency relations, by combining the syntactic and lexical level. In a valid match, while the relation has to be the same ("exact" match), the connected words must be either the same or semantically equivalent in the two languages. For example, *"nsubj (loves, John)"* can match *"nsubj (ama, John)"* and *"nsubj (quiere, John)"* but not *"dobj (quiere, John)"*.

As Algorithm 2 shows, given the dependency tree representations of T

Input: T and H pair represented at the level of syntactic dependency relations
Output: Matching score for each relation r
R = common relations between English and Spanish;
foreach r *in* R **do**
  $Match_r = 0$;
  foreach $DR_r(H)$ **do**
    foreach $DR_r(T)$ **do**
      if $LexMatch(Word_1(H), Word_1(T))$ & $LexMatch(Word_2(H), Word_2(T))$ **then**
        $Match_r = Match_r + 1$;
      end
    end
  end
  $Match_r = \frac{Match_r}{|DR_r(H)|}$;
end

Algorithm 2: Dependency relation matching algorithm

and H, for each grammatical relation (r) we calculate a DR matching score ($Match_r$, see Equation 1) as the number of matching occurrences of r in T and H (respectively $DR_r(T)$ and $DR_r(H)$), divided by the number of occurrences of r in H. Table 3.4 shows a DR example matching between two text portions in English and Spanish.

$$match_r = \frac{|match(DR_r(T), DR_r(H))|}{|DR_r(H)|} \tag{3.1}$$

In our learning framework, DR_match_r values are first calculated for each relation r appearing both in T and H. Then, each value is used as a separate feature, giving the classifier the possibility to learn optimal feature weights from training data.

Overall, this approach resembles the way syntactic information has been used in monolingual textual entailment recognition [Androutsopoulos & Malakasiotis, 2010]. The differences in the proposed adaptation to the

T	**H**
Wolfgang Amadeus Mozart was born in Salzburg	Mozart nació en 1756
1-gr.: PER, was, born, in, LOC	**1-gr.**: PER, nació, en, DATE
2-gr.: PER was, was born, born in, in LOC	**2-gr.**: PER nació, nació en, en DATE
3-gr.: PER was born, was born in, born in LOC	**3-gr.**: PER nació en, nació en DATE
4-gr.: PER was born in, was born in LOC	**4-gr.**: PER nació en DATE
5-gr.: PER was born in LOC	**5-gr.**: -
SPT matching	
1-gr. = 3/4, **2-gr.** = 2/3, **3-gr.** = 1/2	
1-gr.: PER#PER, nació#born, en#in	
2-gr.: PER nació#PER was born, nació en#born in	
3-gr.: PER nació en#PERSON was born in	
4-gr.: -	
5-gr.: -	

Table 3.5: Semantic Phrase Table (SPT) matching between an English text and a Spanish hypothesis.

cross-lingual scenario concerns the use of different dependency parsers for the languages of T and H, and the need to map (manually in our case) the sets of dependency relation labels they output.

In our experiments, in order to extract dependency relation (DR) matching features, the dependency tree representations of English texts and Spanish hypotheses have been produced with DepPattern [Gamallo Otero & Gonzalez Lopez, 2011]. We then mapped the sets of dependency relation labels for the English-Spanish parser output into: Adjunct, Determiner, Object, Subject and Preposition. The dictionary, containing about 9M bilingual word pairs, created during the alignment of the English-Spanish parallel corpora provided the lexical knowledge to perform matches when the connected words are different.

Semantic Phrase Table Matching

Semantic Phrase Table (SPT) matching represents a novel way to leverage the integration of semantics and MT-derived techniques. To this aim, SPT improves CLTE methods relying on pure lexical match, by means of "generalized" phrase tables annotated with shallow semantic labels. Semantically

enhanced phrase tables, with entries in the form "*[LABEL]* $word_1 ... word_n$ *[LABEL]*" (*e.g.* "*[ORG] acquired [ORG]*"), are used as a **recall**-oriented complement to the lexical phrase tables used in machine translation (token-based entries like "*Yahoo acquired Overture*"). The main motivation for this augmentation is that word replacement with semantic tags allows to match T-H tokens that do not occur in the original bilingual parallel corpora used for phrase table extraction. Our hypothesis is that the increase in recall obtained from relaxed matches through semantic tags in place of "out of vocabulary" terms (*e.g.* unseen person, location, or organization names) is an effective way to improve CLTE performance, even at the cost of some loss in precision. Semantic phrase tables, however, have two additional advantages. The first is related to their smaller size and, in turn, its positive impact on system's efficiency, due to the considerable search space reduction. Semantic tags allow to merge different sequences of tokens into a single tag and, consequently, different phrase entries can be unified to one semantic phrase entry. As a result, for instance, the SPT used in our experiments is more than 30% smaller than the original token-based one. The second advantage relates to their potential impact on the confidence of CLTE judgements. Since a semantic tag might cover more than one token in the original entry phrase (*e.g.* "*Wolfgang Amadeus Mozart*" in Table 2, which is covered by the single label "*[PER]*"), SPT entries are often short generalizations of longer original phrases. Consequently, the matching process can benefit from the increased probability of mapping higher order n-grams (*i.e.* those providing more contextual information) from H into T and vice-versa.

Like lexical phrase tables, SPTs are extracted from parallel corpora. As a first step, we annotate the corpora with named-entity taggers for the source and target languages, replacing named entities with general semantic labels chosen from a coarse-grained taxonomy including the categories:

person, location, organization, date and numeric expression. Then, we combine the sequences of unique labels into one single token of the same label, and we run Giza++ [Och & Ney, 2000] to align the resulting semantically augmented corpora. Finally, we extract the semantic phrase table from the augmented aligned corpora using the Moses toolkit [Koehn et al., 2007].

For the matching phase, we first annotate T and H in the same way we labeled our parallel corpora. Then, for each n-gram order (n=1 to 5, excluding punctuation), we use the SPT to calculate a matching score (SPT_match_n, see Equation 3.2), as the number of n-grams in H that match with phrases in T divided by the number of n-grams in H. The matching algorithm is same as Algorithm 1.

$$SPT\_match_n = \frac{|SPT_n(H) \cap SPT(T)|}{|SPT_n(H)|} \tag{3.2}$$

Table 3.5 illustrates SPT matching between two text portions in English and Spanish.

In our learning framework, the computed SPT_match_n scores are used as separate features, giving the classifier the possibility to learn optimal feature weights from training data.

We extracted the semantic phrase table from the augmented corpora in the same way mentioned above for our experiments. We exploited the same parallel corpora mentioned in phrase table extraction phase. The extracted SPT contained about 135M phrase pair entries, which is about 30% smaller than the lexical PT.

Experiments and Results

Accuracy results have been calculated over 800 test pairs of the RTE3-derived CLTE corpus, after training the SVM binary classifier over the 800 development pairs, using different feature sets. We compared our new

Dataset	RTE-3 AVG	Pivot PPT	PT	PT+DR	PT+SPT	PT+SPT+DR
RTE3-derived	62.37%	63.5%	62.6%	63.6%	63.5%	**64.5**%

Table 3.6: CLTE accuracy results over the RTE3 derived dataset.

features with: *i)* the previous CLTE lexical model (PT), *ii)* the best monolingual model (Pivot-PPT) presented in the last section, and *iii)* the average result achieved by participants in the monolingual English RTE-3 evaluation campaign (RTE-3 AVG).

As shown in Table 3.6, also in this case, the best results are achieved using all features (64.5%), while SPT and DR features separately added to PT (PT+SPT, and PT+DR) lead to marginal improvements over the results achieved by the lexical PT (about 1%). This confirms that precision-oriented and recall-oriented features lead to a larger improvement when they are used in combination.

Although extracting and integrating multilingual features in a CLTE learning framework is not always straightforward, the results prove the effectiveness of our combination of lexical evidence with deeper linguistics knowledge. It is worth noting that by using such features, we can also outperform the RTE-3 average score (62.37%) and the best results achieved by exploiting paraphrase tables over the automatic translation of the same dataset into English (63.5%). This further proves the robustness of our proposed cross-lingual feature set in overcoming the noise introduced by the MT component.

In the next chapters, we take advantage of the proposed feature sets dealing with two interesting CLTE applications. We prove that such features can significantly contribute, not only in the theoretical CLTE framework, but also in the cross-lingual application scenarios.

3.5 Summary

This chapter presented the investigations towards cross-lingual Textual Entailment, focusing on possible research directions and alternative methodologies. Feasibility study results have been provided to demonstrate the potentialities of a simple approach that integrates MT and monolingual TE components. As an advanced solution, we approached the cross-lingual Textual Entailment task focusing on the role of lexical knowledge extracted from bilingual parallel corpora.

Our approach builds on the intuition that the vast amount of knowledge that can be extracted from parallel data (in the form of phrase and paraphrase tables) offers a possible solution to the problem. To check the validity of our assumptions we carried out several experiments on an English-Spanish corpus derived from the RTE3 dataset, using phrasal matches as a criterion to approximate entailment. Our results show that phrase and paraphrase tables allow to:

1. Outperform the results achieved with the multilingual lexical resources available.

2. Outperform the average scores obtained by participants in the monolingual RTE-3 challenge.

These improvements can be explained by the fact that the lexical knowledge extracted from parallel data provides good coverage both at the level of single words, and at the level of phrases. We also demonstrated the effectiveness of paraphrase tables as a means to overcome the bias towards single words featured by the existing resources. Finally, we extended the lexical based CLTE methods with a variety of bi-lingual syntactic and semantic features, achieving a considerable improvements.

Overall, our work sets a novel framework for further studies and exper-

iments to improve cross-lingual NLP tasks. In particular, CLTE can be scaled to more complex problems, such as cross-lingual content merging and synchronization, at the same time, contribute to a variety of MT-related tasks, ranging from re-scoring MT outputs to adequacy evaluation.

Chapter 4

Application 1: Entailment-based Multilingual Content Synchronization

4.1 Introduction

The explosion of multilingual user-generated content in websites like Wikipedia provides users with the opportunity to access information about a given topic in their own language. However, to take full advantage of this opportunity, it would be important to present the user with the same content, independently from the language version of the article. Currently, to address this issue, multilingual Wikis rely on contributors to manually translate different pages on the same subject. When contributors update the different language versions independently, translators should separately confront and synchronize each update. This is a demanding task which involves lots of effort, and may create many content dissimilarities and deviations. These problems, which cannot be tackled by asking contributors to adhere to restrictive content creation guidelines, represent an interesting direction for research on automated solutions.

Given two documents about a same topic written in different languages

(*e.g.* Wikipedia articles), we define the *content synchronization* task as the problem of automatically detecting and resolving differences in the information they provide, in order to produce aligned, mutually enriched versions. A roadmap towards the solution of this problem has to take into account a number of challenging subtasks, including:

1. The detection of topically-related portions of the input documents.
2. The identification of information in one page that is novel/more-informative with respect to the content of the other page.
3. The management of contradictions.
4. The translation of novel/more-informative content that has to migrate across documents.
5. The detection of appropriate entry points for integrating the translated material.
6. The generation of readable outputs.

This chapter focuses on the core subtask *2*, setting it as an application-oriented, cross-lingual variant of the Textual Entailment (TE) recognition task Dagan & Glickman [2004]. Along with this direction, we define and conduct experiments with cross-lingual textual entailment in a real application scenario. By now, cross-lingual textual entailment (CLTE) has only been applied to available (monolingual English) TE datasets (in the previous chapter), transformed into their cross-lingual counterpart by translating the hypotheses into other languages (*e.g.* from English into Spanish). In the previous chapter, no experiments had been conducted on a datasets with different notion, or in an application-oriented framework. In this framework, our experiments are carried out over the only dataset acquired

to represent the multilingual content synchronization scenario (will be discussed in detail in Chapter 6) which arises a richer inventory of phenomena [Negri et al., 2012].

4.2 CLTE-based Content Synchronization

Currently, multilingual Wikis rely on users to manually translate different Wiki pages on the same subject. This is not only a time-consuming procedure but also the source of many inconsistencies, as users update the different language versions separately, and every update would require translators to compare the different language versions and synchronize the updates. The goal of automatic content synchronization system is to identify content discrepancies across different language versions of Wiki pages, and merge them to produce synchronized versions.

The content synchronization system integrates the Structural Analysis (SA), Machine Translation (MT) and Cross-Lingual Textual Entailment (CLTE) technologies in a three-step process where:

1. SA analyzes the structure of the input Wiki pages, automatically identifying segments that represent semantically coherent portions (paragraphs, sentences or chunks).

2. CLTE identifies text portions that should "migrate from one page to the other.

3. MT translates these portions in the appropriate target language.

Figure 4.1 shows a schematic representation of such system. The entailment-based content merging component is in charge of annotating the input pages in terms of: *i)* overlapping information that does not need to be translated for synchronization, and *ii)* information that has to be

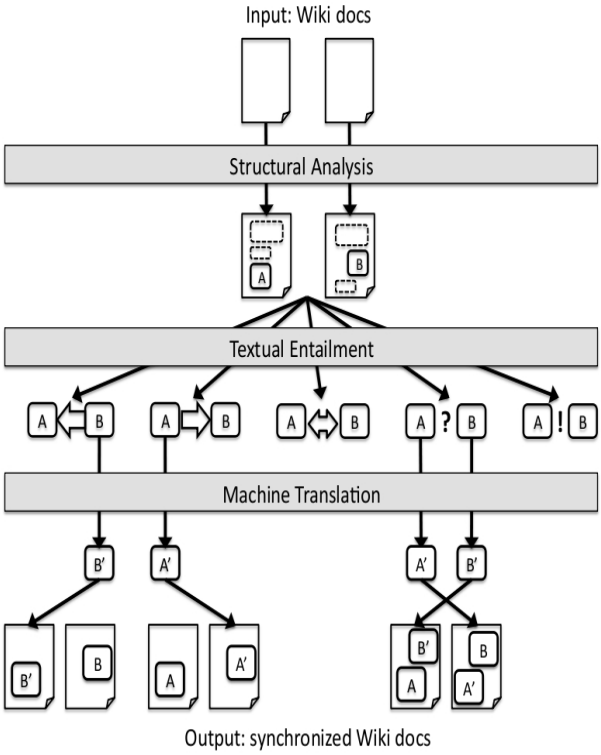

Figure 4.1: A framework for automatic content synchronization of multilingual Wiki content.

translated and has to migrate from one page to the other (*i.e.* more specific information, or factual information that is present only in one page). The output of this component will allow the MT component to focus on translating content that is novel with respect to the Wiki page into which translated content is to be inserted. In terms of entailment checking, Figure 4.1 depicts all the possible relations between two topically related text fragments (A and B). The first two cases (marked as A←B and A→B) respectively indicate situations where a text portion is entailed (*i.e.* is more general) or entails (*i.e.* is more specific than) the other. In this case, the fragment providing more specific information will be translated and inserted in the other page in the appropriate place. The third case (A↔B) indicates semantically equivalent text portions which will be left untouched. The fourth case (A ? B) indicates situations where the two text

fragments neither entail nor contradict each other. Handled as fragments providing novel information, both of them will be translated and inserted in the other page in the appropriate place. The fifth case (A ! B) represents the situation where the two topically related texts contradict each other. In principle, the contradiction should be solved, the correct information kept, and shared by the two pages.

Inline with the focus of this thesis, we believe that the adoption of entailment-based techniques to address content synchronization looks promising, as one of the main components of this task can be formalized as an entailment-related problem. Explaining the entailment-based approach in Figure 4.1 by real world examples, given two pages (*P1* and *P2*), issues include identifying, and properly managing[1]:

- Text portions in *P1* and *P2* that express exactly the same meaning (bi-directional entailment, or semantic equivalence, as in: *"Mozart was born in Salzburg, Austria"*↔*"Mozart was born in the Austrian city of Salzburg"*). In such cases, since there is no information that has to migrate across *P1* and *P2*, the two text portions will remain the same.

- Text portions in *P1* that are more informative than portions in *P2* (forward entailment from *P1* to *P2*, as in: *"Mozart was born in Salzburg"*→*"Mozart was born in Austria"*). In such cases, the entailing (more informative) portion from *P1* has to be translated and migrated to *P2* in order to replace the entailed (less informative) fragment;

- Text portions in *P2* that are more informative than portions in *P1*

[1]For the sake of clarity, the examples provided in this section involve simple English sentences. Although the entailment-based approach is also suitable for the monolingual scenario, the experiments reported in the remainder of this chapter are carried out on the English/German dataset.

(backward entailment from *P2* to *P1*), and should be translated to replace them;

- Text portions in *P1* describing facts that are not present in *P2*, and vice-versa (the "unknown" cases in RTE parlance, as in: *"Mozart was born in Salzburg"*−?− *"Mozart was born in 1756"*). In such cases, the novel information from both sides has to be translated and migrated in order to mutually enrich the two pages.

- Meaning discrepancies between text portions in *P1* and text portions in *P2* ("contradictions" in RTE parlance, as in: *"Mozart was born in Salzburg"*−!− *"Mozart was born in Wien"*).

Our framework presents two main differences with respect to the standard formulation of the entailment recognition task (as it is adopted, for instance in the previous chapter). First, in the RTE scenario only unidirectional entailment relations between texts and hypotheses are considered. In contrast, content synchronization requires to capture entailment relations in all possible directions. Second, targeting the synchronization of documents in different languages, our scenario adds multilinguality issues to the complexity of semantic inference at the textual level.

So far, despite its many potential applications, multi-directional TE recognition has been addressed (in the very recent NTCIR-9 RITE Multi-class subtask[2]), at the monolingual level. However, we proposed the task of cross-lingual content synchronization scenario in the most recent Semantic Evaluation (SemEval) series of workshops which focuses on the evaluation of semantic analysis systems,[3] as one of the tasks which can bring the MT and Semantics community closer. We believe that this task can rise the challenge of dealing with a real world entailment task at the

[2] http://artigas.lti.cs.cmu.edu/rite/
[3] http://www.cs.york.ac.uk/semeval-2012/task8/

multilingual scenario.

4.3 Experiments

To demonstrate the effectiveness of our approach using different feature sets proposed in Chapter 3, we present in this section our experimental settings, the dataset we used, and different settings adopted for such scenario. Our experiments aims at: *i)* proving that CLTE represents a viable solution to detect semantic equivalence and information disparity for multilingual content synchronization, and *ii)* verifying if lexical, semantic and syntactic features can jointly contribute to improve the CLTE results obtained by using lexical phrase tables.

4.3.1 Dataset

In order to cope with the necessity of having a multilingual content synchronization dataset, we developed a "divide and conquer" methodology based on crowdsourcing [Negri et al., 2011]. This aimed at creating a CLTE corpora from scratch by decomposing a complex content generation task in a pipeline of simpler subtasks accessible to a large crowd of non-experts. The quality control mechanisms were also integrated at each stage of this process. In this case, a complex multilingual task is reduced to a sequence of simpler subtasks where the most difficult one, the generation of entailment pairs, is entirely monolingual. Besides ensuring cost-effectiveness, our solution allowed to overcome the problem of finding workers that are proficient in multiple languages.

The result of this work is the first and only available dataset containing both monolingual and cross-lingual corpora for several combinations of

texts-hypotheses in English, Italian, and German. Among the advantages of this method it's worth mentioning: *i)* the full alignment between the created corpora, *ii)* the possibility to extend the dataset to new languages by simply crowdsourcing the translation of English sentences, and *iii)* the possibility to create a corpora for content synchronization task, featuring more complex entailment relations than the traditional ones. The last chapter of this thesis (Chapter 6) explains our strategies in creating this dataset in detail.

In our experiments, we used this corpus, which contains both monolingual and cross-lingual aligned pairs in several combinations of English, Italian and German, annotated with multi-directional entailment relations. This dataset contains 500 pairs for each language combination, which we equally divided into training and test sets. Each pair in the dataset is annotated with "Bidirectional", "Forward", or "Backward" entailment judgements. Although highly relevant for the overall content synchronization task, "Contradiction" and "Unknown" cases (*i.e.* "NO" entailment in both directions) are not present in the annotation.

We chose the English-German (ENG-GER) portion of the dataset since, compared to the others, for such language pair MT systems performance is often lower. This makes the adoption of simpler solutions based on the pivoting approach, proposed earlier in [Mehdad et al., 2010b] (Chapter 3), more vulnerable. Besides the intrinsic difficulty of the task, the obstacle represented by the noise introduced by translations in the resulting (monolingual) entailment pairs further motivates the use of an integrated approach to CLTE, as proposed earlier in [Mehdad et al., 2011] (Chapter 3).

4.3.2 Features

Aiming at entailment-based content synchronization, we explore the potential contribution of lexical, syntactic and semantic features, as it was proposed in Chapter 3, in a supervised learning framework. Our model builds on three main feature sets, respectively derived from: *i)* phrase tables, *ii)* dependency relations, and *iii)* semantic phrase tables.

1. **Phrase Table (PT) matching:** through these features, a semantic judgement about entailment is made exclusively on the basis of lexical evidence. To build the English-German phrase tables for matching, we combined the Europarl, News Commentary and de-news[4] parallel corpora. After tokenization,[5] Giza++ [Och & Ney, 2000] and Moses [Koehn et al., 2007] were respectively used to align the corpora and extract a lexical phrase table (PT). This resulted a phrase table with about 45M phrase pair entries.

2. **Dependency Relation (DR) matching** targets the increase of CLTE precision. Adding syntactic constraints to the matching process, DR features aim to reduce the amount of wrong matches often occurring at the lexical level. Dependency relations (DR) were extracted running the Stanford parser [Rafferty & Manning, 2008; De Marneffe et al., 2006]. We then mapped the sets of dependency relation labels for English-German parser output into: adjective, adverb (modifier), verb (root), subject, object, numeral, conjunction, and modal verbs. The dictionary created during the alignment of the parallel corpora provided the lexical knowledge to perform matches when the connected words are different, but semantically equivalent in the two languages. The method for extracting the features were

[4]Available at http://homepages.inf.ed.ac.uk/pkoehn/publications/de-news/
[5]With the standard tokenizer released with Moses.

explained in Chapter 3 in details.

3. **Semantic Phrase Table (SPT) matching:** aims at improving CLTE methods relying on pure lexical match, by means of generalized phrase tables annotated with shallow semantic labels (as it was discussed in Chapter 3). For creating the semantic phrase table (SPT) we used the Stanford named entity tagger [Faruqui & Padó, 2010; Finkel et al., 2005] to annotate with semantic tags the parallel corpora for English and German. Then, we combined the sequences of unique labels into one single token of the same label, and we run Giza++ [Och & Ney, 2000] to align the resulting semantically augmented corpora. Finally, we extracted the semantic phrase table from the augmented aligned corpora using the Moses toolkit [Koehn et al., 2007]. This process created a SPT containing about 35M phrase pair entries, which is about 20% smaller in size than the lexical PT.

To combine and weight features at different levels (PT, SPT and DR), we used a Support Vector Machine (SVM) classifier, SVMlight [Joachims, 1999a].

4.3.3 Evaluation settings

In order to experiment under testing conditions of increasing complexity, we set the CLTE problem both as a two-way and as a three-way classification task. Two-way classification casts multi-directional entailment as a unidirectional problem, where each pair is analyzed checking for entailment both from left to right and from right to left (with "Yes" and "No" as possible entailment judgements). To this aim, the pairs representing the different types of entailment relations have been duplicated as follows:

1. Bi-directional entailment examples ($T \leftrightarrow H$) have been duplicated into: *i)* a positive forward entailment pair where T entails H ($T \rightarrow$

H), and *ii)* a positive backward entailment pair where H entails T ($T \leftarrow H$).

2. Forward entailment examples ($T \rightarrow H$) have been duplicated into: *i)* a positive forward entailment pair where T entails H ($T \rightarrow H$), and *ii)* a negative backward entailment pair where H does not entail T ($T \not\leftarrow H$).

3. Backward entailment examples ($T \leftarrow H$) have been duplicated into: *i)* a negative forward entailment pair where T does not entail H ($T \not\rightarrow H$), and *ii)* a positive backward entailment pair where H entails T ($T \leftarrow H$).

Two-way classification represents an intuitive solution to capture multi-directional entailment relations but, at the same time, a suboptimal approach in terms of efficiency since two checks are performed for each pair. Three-way classification is more efficient since it does not require the combination of independent unidirectional judgements, but at the same time more challenging due to the higher difficulty of multiclass learning (with "Forward", "Backward", and "Bidirectional" as possible judgements), especially with small datasets.

4.3.4 Results

Accuracy results for different feature sets have been calculated over 250 test pairs (duplicated into 500 in the 2-way classification experiments) of the ENG-GER content synchronization corpus. We carried out three types of evaluation, whose results are reported in Table 4.3.4. The first one is a *lenient* 2-way accuracy score ("Lenient" column) that considers the percentage of correctly classified pairs ("YES" or "NO") out of the total duplicated test pairs. The second evaluation method is a more *strict* synchronization-oriented accuracy ("CS column") where each original test

example is correctly classified if both pairs originated from it are correctly judged (*i.e.* "YES-YES" for bidirectional, "YES-NO" for forward, and "NO-YES" for backward entailment). Strict evaluation scores are also split into "bidir" (corresponding to the performance on bidirectional entailment pairs), and "unidir" (corresponding to forward and backward entailment annotation). This aims at checking: *i)* the performance of each feature set in detecting the direction of entailment, and *ii)* the possibility to tune the SVM classifier, optimizing results for one of the two classes, still keeping overall CS performance under control.

Finally, the last column ("3-way") presents the most challenging scenario where the SVM model learns to classify the test pairs based on the exact class ("Bidirectional", "Forward", and "Backward") using a multi-class classifier [Crammer & Singer, 2002]. We also compare our results with two pivoting approaches, checking for entailment between the original English texts and the translated German hypotheses.[6] The first (Pivot-EDITS), uses an optimized distance-based model implemented in the open source RTE system EDITS [Kouylekov et al., 2011]. To obtain the optimal model, we run the EDITS-GA over the training set to automatically find the best settings and algorithm for such dataset using the genetic algorithm discussed in Chapter 2. The resulting model (among all others) was based on using token edit distance algorithm by removing the stop-words from each pair.

The second (Pivot-PPT) exploits paraphrase tables for phrase matching, and represents the best monolingual model presented in Mehdad et al. [2011], discussed in Chapter 2.

Table 4.1 demonstrates the success of our results in proving the two main claims of this chapter.

1. On both 2-way and 3-way classification, all the feature sets outper-

[6] Using Google Translate.

Feature sets	2-way							3-way
	Lenient	tuned for unidir			tuned for bidir			
		CS	unidir	bidir	CS	unidir	bidir	
PT	77.4	57.8	61.3	50.0	57.0	52.8	66.2	57.4
PT+DR	77.6	58.6	60.7	54.1	58.2	57.7	59.5	57.8
PT+SPT	79.5	62.4	64.4	58.1	59.5	57.7	63.5	58.7
PT+SPT+DR	79.8	**63.3**	66.9	55.4	**60.3**	59.5	63.2	**61.6**
Pivot-EDITS	72.8	27.4	14.7	55.4	27.4	14.7	55.4	25.3
Pivot-PPT	80.7	57.0	46.0	81.1	57.0	46.0	81.1	56.1

Table 4.1: CLTE for content synchronization accuracy results. Three types of evaluation, are reported: *lenient 2-way* classification, ("YES" and "NO" judgements), *combined 2-way* classification ("YES-YES" for bidirectional, "YES-NO" for forward, and "NO-YES" for backward), and *3-way* classification ("Bidirectional", "Forward" and "Backward"). Different CLTE models are compared with two pivoting approaches (Pivot-EDITS, and Pivot-PPT).

form the approaches taken as terms of comparison. The 61.6% accuracy achieved in the most challenging setting (3-way) demonstrates the effectiveness of our approach to capture meaning equivalence and information disparity in cross-lingual texts.

2. Syntactic and semantic features, combined with lexical features (PT+ SPT+DR) significantly improve[7] the CLTE state-of-the-art lexical model (PT), for all experimental settings (2 way and 3-way).

A further analysis of the reported results brings to other interesting observations.

- Semantic phrase table matching (PT+ SPT) constantly improves lexical phrase table matching (PT) for all settings (Lenient, CS and 3-way) and parameters (unidir/bidir tuning). Such improvement can

[7] $p < 0.05$, calculated using the approximate randomization test implemented in Padó [2006].

be motivated by the increased coverage of SPTs (matching more and longer n-grams), and the consequent recall improvement over PTs.

- DR always helps in boosting PT matching results (PT+DR and PT+ SPT+DR). Such improvement is likely due to the gain in precision brought by syntactic constraints. However, the increase over the lexical phrase table matching (PT+DR) is minimal. This might be due to the fact that both PT and DR features are precision-oriented, and their effectiveness becomes evident only in combination with recall-oriented features (*e.g.* SPT).

- The cross-lingual models can be tuned to obtain better results for bidirectional (bidir) or unidirectional (unidir) entailment with minimal or zero loss in the overall accuracy (CS). This is potentially helpful in the scenarios where: *i)* a dataset is unbalanced and biased towards a class, or *ii)* there is a need to boost bidirectional or unidirectional entailment recognition (semantic equivalence vs. RTE-like entailment).

- The high results in the RTE-like setting (Lenient 2-way classification), ranging from 77% to 80% are above the state-of-the-art in monolingual RTE. This is not surprising considering that duplicating the original pairs into "YES" and "NO" creates an unbalanced dataset with a higher number of "YES" pairs (around 65%). However, the fact that lenient judgements represent a relatively easier task, does not reduce the difficulty of the multi-directional CLTE task here proposed.

Further interesting observations emerge from the comparison with the results achieved on monolingual data by the two pivoting approaches.

- When dealing with MT-derived inputs, there is a drop in the overall results. In other words, cross-lingual models outperform the pivoting models significantly. This suggests that the noise introduced by

incorrect translations makes the pivoting approach less attractive in comparison with the more robust cross-lingual models.

- The accuracy obtained in the lenient evaluation using paraphrase tables PPT (80.7%) is minimally better than using the cross-lingual model (79.8%), however it does not hold in other cases (*e.g.* CS or 3-way). This demonstrates that monolingual models can somehow cope with the traditional entailment judgements, while they lack significantly in judging the direction of the entailment in more challenging scenarios. This negative impact, especially in EDITS, might be due to the fact that available algorithms often rely on similarity-based methods, that work reasonably well only with bidirectional cases. This also emphasizes the need of more RTE datasets addressing real world application scenarios.

- The monolingual models are not easily tunable, and the attempts to tune such models drive to a drop in bidirectional cases with no improvement in unidirectional pairs. This further proves the effectiveness of our cross-lingual models in approaching this task.

4.4 Open Issues and Future Directions

Although relevant for the content synchronization task, "contradictions" and "unknown" cases (*i.e.* "NO" entailment in both directions) are not considered at this stage of our work, and are left as a future research direction. On one side, contradictions would require to decide which of the two elements of a pair provides true, or more reliable information. Such additional level of complexity is currently out of the scope of our research, which builds on the assumption that both statements provide true information. On the other side, unknown cases are not represented

yet in the CLTE datasets, since the corpus creation methodology adopted did not target the collection of such kind of entailment pairs.

As a step towards this, we proposed the "*Cross-lingual Textual Entailment for Content Synchronization*" task in SemEval 2012, adding "*No Entailment*" (T1 ! → T2 and T1 ! ← T2)[8] pairs to the evaluation scenario. This larger dataset consists of 1,000 CLTE pairs (500 for training and 500 for test), balanced with respect to the four entailment judgments (bidirectional, forward, backward, and no entailment). The dataset was created following the same crowdsourcing-based methodology, that will be discussed in Chapter 6 [Negri et al., 2011], which consisted of the following steps:

1. English sentences were selected from copyright-free sources, *i.e.* Wikipedia and Wikinews, and represent T1 in the entailment pair.

2. Each T1 was modified through crowdsourcing in various ways (*e.g.* introducing lexical and syntactic changes, adding and removing portions of text, etc.) in order to obtain a corresponding T2.

3. Each T1 was paired to the corresponding T2, and the resulting pairs were annotated with the entailment judgment. The final result was a monolingual English dataset.

4. In order to create the cross-lingual datasets, each English T1 was translated into different languages (*i.e.* Spanish, German, Italian and French).

5. By pairing the translated T1 with the corresponding T2 in English, four cross-lingual datasets were obtained.

[8] T1 and T2 represent the first and second elements of the pair (*i.e.* T and H in the traditional RTE scenario).

6. The overall final result is a multilingual parallel entailment corpus, where T1's are in 5 different languages (*i.e.* English, Spanish, German, Italian, and French), and T2's are in English.

7. To ensure the quality of the dataset, all the pairs were manually checked by two expert annotators and modified where necessary.

Our future work will address both the content synchronization, and cross-lingual textual entailment. On one side, we plan to explore different features using various dimensions to improve our CLTE model, and consequently the content synchronization results. One possible direction is to consider topic modelling to measure the relatedness of the texts. It is worth mentioning that we tried few approaches to exploit some information from a bilingual LSA model, yet there is no significant improvement in such direction. Another interesting direction is to investigate the potential of wikipedia entity linking based features as a semantic similarity measure to boost the performance. On the other side, we explore the possibility of adopting our feature sets in order to deal with the "unknown" cases present in the new dataset, in order to drive to a more realistic evaluation scenario [Mehdad & Negri, 2012].

4.5 Summary

In this chapter we addressed *multilingual content synchronization*, which represents at the same time a challenging application scenario for a variety of NLP technologies, and a shared research framework for the joint contribution of semantics and MT technology. Our first step towards the success of this endeavour consists in formalizing the core aspects of the task as a cross-lingual textual entailment (CLTE) problem. Towards this direction, we took a step further applying an improved CLTE model, and

providing the successful results over different settings. Building on our previous works targeting *(i)* the investigation of possible approaches to CLTE, and *(ii)* the collection of parallel CLTE datasets, this chapter took a step further applying an improved CLTE model, and providing the successful results over the only content synchronization dataset available. Along with the our proposed approaches and the collected corpora, our results represent a strong element to build a solid framework for this new research direction [Mehdad et al., 2012a].

Chapter 5

Application 2: Evaluating the Adequacy of Machine Translation Output without References

5.1 Introduction

While syntactically informed modelling for statistical MT is an active field of research that has recently gained major attention from the MT community, work on integrating semantic models of adequacy into MT is still at preliminary stages. This situation holds not only for system development (most current methods disregard semantic information, in favour of statistical models of words distribution), but also for system evaluation. To realize its full potential, however, MT is now in the need of semantic-aware techniques, capable of complementing frequency counts with meaning representations.

In the effort of pushing semantics into MT technology, in this chapter we focus on the evaluation dimension. Restricting our investigation to some of the more pressing issues emerging from this area of research, we focus on: *i)* an automatic evaluation method that avoids the use of reference translations, and *ii)* a method for evaluating translation adequacy.

Our approach builds on the core advancements in cross-lingual textual entailment (CLTE) recognition, which provides a natural framework to address MT adequacy [Mehdad et al., 2012b]. In particular, we cast the problem as a CLTE task where bi-directional entailment between source and target is considered as evidence of translation adequacy. Besides avoiding the use of references, the proposed solution differs from most previous methods which typically rely on surface-level features, often extracted from the source or the target sentence taken in isolation (*e.g.* "average length of source sentence words").

Although some of these features might correlate well with adequacy, they capture semantic equivalence only indirectly, and at the level of a probabilistic prediction. Focusing on a combination of surface, syntactic and semantic features, extracted from *both* source and target (*e.g.* "source-target length ratio", "dependency relations in common"), our approach leads to informed adequacy judgements derived from the actual observation of a translation *given the source sentence*. Our method shows high correlation with human judgements and good results on different datasets and evaluation settings, without relying on reference translations.

5.2 MT evaluation

Machine translation (MT) can be defined as a task for automatically transforming texts in one language into texts in another language, producing fluent output texts that preserve the meaning of the source input texts. Statistical MT (SMT) has recently achieved significant progress in modelling the fluency and adequacy of translations as measured by commonly used automated evaluation metrics.

MT evaluation, especially by means of automatic metrics, serves different purposes:

- Detecting and analyzing possible errors and possibly determining the sources that cause them. This could improve the system significantly and provide better translation at the end of each development cycle.
- Ranking alternative MT systems or different versions of the same system to systematically evaluate their cumulative development.
- Optimizing and tuning MT systems by fixing their configurations and parameters in such a way to achieve the best performance.

While manual (human) evaluations are informative and usually of higher quality, they demand a costly procedure. Moreover, they are subjective, not replicable and not reusable. However, automatic evaluation methods are often efficient, objective and re-usable.

Several automatic metrics, based on different similarity criteria and levels, have been proposed and used in the past. These metrics are mainly based on comparisons between automatic and human reference translations. Most of these metrics score the MT output versus human translation references using different lexical similarities based on: *i)* edit distance (*e.g.* TER [Snover et al., 2006], WER [Nießen et al., 2000] and PER [Tillmann et al., 1997]), *ii)* precision (*e.g.* BLEU [Papineni et al., 2002] and NIST [Doddington, 2002]). *iii)* recall (*e.g.* ROUGE [Lin, 2003]), and *iv)* precision and recall (*e.g.* GTM [Melamed et al., 2003] and METEOR [Banerjee & Lavie, 2005]).

Such measures, especially BLEU, have been widely adopted by the MT community. However, due to the variability of natural languages in terms of possible ways to express the same meaning, reliable lexical similarity metrics depend on the availability of costly, hand-crafted different realizations of the same source sentence in the target language. Moreover, such metrics do not not consistently reward translation adequacy. In order to overcome such shortcomings, some recent works proposed the metrics that

are able to approximately assess meaning equivalence between candidate and reference translations. Among these, Giménez & Màrquez [2007] proposed a heterogeneous set comprising overlapping and matching metrics, compiled from a rich set of variants at five different linguistic levels: lexical, shallow-syntactic, syntactic, shallow-semantic and semantic. More similar to our approach, Padó et al. [2009] proposed semantic adequacy metrics that exploit feature representations motivated by textual entailment. Both metrics, however, highly depend on the availability of multiple reference translations.

Despite the vast growth of automatic metrics for MT evaluation and coping with some imperfections of this technology, there are still several problems in the current methodology for MT evaluation:

1. There is a large drop in automatic metric's performance when there is a lack in availability of the reference translations. As it mentioned earlier, the quality of such metrics is highly dependant on the number of reference translations which are prepared by human. This makes such methods incompetent when there is a shortage in time or finance.

2. It is often not easy to interpret such measures to a meaningful scale in order to get some insight from. For example it is difficult to answer *"what does it mean when the BLEU score is 0.04"*.

3. The lack of information about the quality of a MT system output (even in the case of post-editing), which could be relevant and interesting for human translators, has been always an issue for automatic metrics. Moreover, providing some information that can reveal the capability of a MT system in providing an acceptable translation is not fully explored.

4. The lack of semantic information in MT evaluation and MT systems,

specifically at the multilingual level, has grown up MT systems illiterate, in terms of semantics and meaning. Since translation is very much influenced by language variability, semantic aware models are needed for MT evaluation. Because of the complexity of SMT algorithms, it is not straightforward to embed semantic knowledge. However, the automatic evaluation process can be a good framework to integrate such features in MT technology.

This arises the need to overcome the mentioned problems through the development of systems and algorithms which can judge the adequacy of MT output (*i.e.* problems 2 and 3) without the need of reference translations (*i.e.* problem 1), which are enriched by semantic information (*i.e.* problem 4). Without more suitable approaches to address these difficulties, the introduction of semantics in MT technology is far to be achieved. Fortunately, CLTE technology can benefit such application by integrating lexical-semantic knowledge in MT evaluation. We believe that CLTE could improve MT evaluation directly, besides indirectly favour the improvement of MT approaches.

5.3 Predicting MT Adequacy

Evaluating the MT output exposes different dimensions for further exploration. Fluency, adequacy and quality are among the most relevant features to be investigated in order to evaluate the MT system output. Each dimension explores different characteristics of the translated sentences, which ideally reflect the weakness and strength of MT systems.

Fluency mainly embeds the naturalness of the output sentence. In other words, it reflects how the MT output can be read like a sentence written by a native speaker. This criterion can be evaluated separately, disregarding other characteristics of the output such as meaning or under-

standability. However, an output can be totally fluent but very different in terms of the meaning and information from the source sentence. The features used for evaluating fluency are mainly source-independent, focusing on the grammatical and readability characteristics of the target.

Adequacy is explained as a translation characteristic that preserves the meaning of the source text without adding/removing any information to/from it. Intuitively, this criterion is related to the semantics and content of the output rather than its grammaticality or readability. However, it is often very challenging to draw a precise borderline between these two criteria, considering the nature of natural languages. This difficulty is observed in several MT evaluation campaigns [Callison-Burch et al., 2010]. The features used for evaluating adequacy are mainly source-target dependent, focusing on the meaning and information present in both source and target.

Quality estimation (QE) focuses mainly on assessing the quality of the output without distinguishing between fluency and adequacy. The focuses of such measures are mainly on the acceptably of the output sentences/segments, or the amount of post editing needed to achieve a good translation. The features that are used for evaluating the quality are a combination of source, target and both source and target, focusing on various characteristics of the output (*e.g.* "source complexity" and "target fluency").

Moving toward having an automatic measure addressing the adequacy of MT output and overcoming the problems mentioned in the previous section (*e.g.* need of many reference translations), lead us to focus on the adequacy evaluation without the use of reference translations. Early attempts to avoid reference translations addressed *quality estimation* (QE) by means of large numbers of source, target, and system-dependent features to discriminate between "good" and "bad" translations (*e.g.* Blatz et al. [2004];

Quirk [2004]). More recently Specia et al. [2009]; Specia & Farzindar [2010]; Specia [2011] conducted a series of experiments using features designed to estimate translation post-editing effort (in terms of volume and time) as an indicator of MT output quality. Good results in QE have been achieved by adding linguistic information such as shallow parsing, POS tags [Xiong et al., 2010], or dependency relations [Bach et al., 2011; Avramidis et al., 2011] as features. However, in general these approaches do not distinguish between fluency (*i.e.* syntactic correctness of the output translation) and adequacy, and mostly rely on fluency-oriented features (*e.g.* "number of punctuation marks"). As a result, however, a simple surface form variation is given the same importance of a content word variation that changes the meaning of the sentence. To the best of our knowledge, only Specia et al. [2011] proposed an approach to frame MT evaluation as an adequacy estimation problem. However, their method still includes many features which are not adequacy focused, and often look either at the source or at the target in isolation (see for instance "source complexity" and "target fluency" features). Moreover, the actual contribution of the adequacy features used is not always evident and, for some testing conditions, marginal.

Our approach to adequacy evaluation builds on and extends the mentioned works, taking advantage of CLTE framework. Similarly to [Padó et al., 2009] we rely on the notion of textual entailment, but declined in its cross-lingual sense in order to bypass the need of reference translations. Similarly to Blatz et al. [2004]; Quirk [2004], we try to discriminate between "good" and "bad" translations, by focusing exclusively on adequacy. To this aim, similarly to Xiong et al. [2010], Bach et al. [2011], Avramidis et al. [2011], and Specia et al. [2009, 2011] we investigate a large set of features, but limited to source-target dependent ones (see Table 5.1).

QE-oriented features (Specia et al. 2010b)	**adequacy**	**fluency**	**src**	**tgt**
avg. word length & avg # of translations	?	?	x	
n-gram frequencies	?	x	x	
bracket & quotation mismatching & POS LM probabilities		x		x
LM probabilities		x	x	x
alignment score	x	x	x	x
length & ratio	x		x	x
brackets, punctuations, numbers and content/non-content words	x		x	x
Adequacy-oriented features				
words, punctuations and OOV match and ratio (F)	x		x	x
POS tags match and ratio (Syn)	x		x	x
syntactic roles match and ratio (SSyn)	x		x	x
dependency relations match (DR)	x		x	x
phrase table match (PT)	x		x	x
semantic-aware phrase table match(SPT)	x		x	x

Table 5.1: Comparison between our *adequacy-oriented* features and the *quality* features (QE) used by Specia et al. [2009], in terms of source/target derivation and adequacy/fluency nature.

5.4 CLTE for adequacy evaluation

We address adequacy evaluation by relying on cross-lingual textual entailment recognition as a way to measure to what extent a source sentence and its automatic translation are semantically similar. CLTE, as it has been proposed and discussed in Chapter 3, is an extension of textual entailment [Dagan & Glickman, 2004] that consists in deciding, given a text T and a hypothesis H *in different languages*, if the meaning of H can be inferred from the meaning of T.

The main motivation in approaching adequacy evaluation using CLTE is that an adequate translation and the source text should convey the same meaning. In terms of entailment, this means that an adequate MT output and the source sentence should entail each other (bi-directional entailment). Losing or altering part of the meaning conveyed by the source sentence (*i.e.* having more, or different information in one of the two sides) will change the entailment direction and, consequently, the adequacy judgement. Framed

in this way, CLTE-based adequacy evaluation methods can be designed to distinguish meaning-preserving variations from true divergence, regardless of reference translations. Moreover, considering only the semantics of the source (T) and the target (H), CLTE-based adequacy judgements are by definition fully independent from fluency and grammaticality issues.

Similarly to many monolingual TE approaches, CLTE solutions proposed so far adopt supervised learning methods, with features that measure to what extent the hypotheses can be mapped into the texts. The underlying assumption is that the probability of entailment is proportional to the number of words in H that can be mapped to words in T (as it was explained in Chapter 3). Such mapping can be carried out at different word representation levels (*e.g.* tokens, lemmas, stems), possibly with the support of lexical knowledge in order to cross the language barrier between T and H (*e.g.* dictionaries, phrase tables). Under the same assumption, since in the adequacy evaluation framework the entailment relation should hold in both directions, the mapping is performed both from the source to the target and vice-versa, building on features extracted from both sentences. Moreover, to improve over previous CLTE methods and boost MT adequacy evaluation performance, we explore the joint contribution of a number of linguistically motivated features.

5.4.1 Features

Aiming at objective adequacy evaluation, our method limits the recourse to MT system-dependent features to reduce the bias of evaluating MT technology with it's own core methods. The experiments described in the following sections are carried out on publicly available English-Spanish datasets, exploring the potential of a combination of surface, syntactic and semantic features. Language-dependent features are extracted by exploiting a number of tools for the two languages (part-of-speech taggers,

dependency parsers and named entity recognizers). Our feature set can be described as follows:

- **Surface Form (F)** features consider the number of words, punctuation marks and non-word markers (*e.g.* quotations and brackets) in source and target, as well as their ratios (source/target and target/source), and the number of out of vocabulary terms encountered.

- **Shallow Syntactic (SSyn)** features consider the number and ratios of common part-of-speech (POS) tags in source and target. Since the list of valid POS tags varies for different languages, we mapped English and Spanish tags into a common list using the FreeLing tagger [Carreras et al., 2004]. Our common POS list for English-Spanish language pair is: Noun, Verb, Adjective, Adverb, Number, Pronoun, Conjunction, Punctuation, Preposition and Symbol.

- **Syntactic (Syn)** features consider the number and ratios of dependency roles common to source and target. To create a unique list of roles, we used the DepPattern [Gamallo Otero & Gonzalez Lopez, 2011] package, which provides English and Spanish dependency parsers. Our common dependency roles are: Adjunct, Determiner, Object, Subject, Preposition and Root.

- **Dependency Relation (DR)** matching features capture similarities between dependency relations, combining syntactic and lexical levels. DR features were extracted in the same way discussed in Chapter 3.

  Term matching is carried out by means of a bilingual dictionary extracted from parallel corpora, as described in the next paragraph. Given the dependency tree representations of source and target produced with DepPattern, for each grammatical relation r we calculate two DR matching scores as the number of matching occurrences of r

in both source and target, respectively normalized by: *i)* the number of occurrences of r in the source, and *ii)* the number of occurrences of r in the target.

Overall, this approach resembles other syntax-based methods previously proposed for MT evaluation at the monolingual level Giménez & Màrquez [2007]. Its adaptation to the cross-lingual scenario, however, is less straightforward for at least two reasons. First, dependency parsers for different language combinations is not always trivial and parsing the noisy output of MT systems is challenging. Second, the alignment of different syntactic representations of T and H requires some additional effort (manual in our case) to define mapping rules for the relations of interest.

- **Phrase Table (PT)** matching features are calculated as described in Chapter 3, with a phrasal matching algorithm that takes advantage of a lexical phrase table extracted from a bilingual parallel corpus. The algorithm determines the number of source phrases (1 to 5-grams, at the level of tokens, lemmas and stems) that can be mapped into target word sequences, and vice-versa. To build our English-Spanish phrase table, we used the Europarl, News Commentary and United Nations Spanish-English parallel corpora. After tokenization, the Giza++ Och & Ney [2000] and the Moses toolkit Koehn et al. [2007] were respectively used to align the corpora and extract the phrase table. The resulted PT contained 200M phrase pair entries. Although the phrase table was generated using MT technology, its use to compute our features is still compatible with a system-independent approach since the extraction is carried out without tuning the process towards any particular task. Moreover, our phrase matching algorithm integrates matches from overlapping n-grams of different size and nature (tokens,

lemmas and stems) which current MT decoding algorithms cannot explore for complexity reasons.

- **Semantic Phrase Table (SPT)** matching features are calculated using phrase tables annotated with shallow semantic labels. SPTs have been extracted from the same parallel corpora used to build lexical PTs. To this aim, we first annotated the corpora with the FreeLing named-entity tagger, replacing named entities with general semantic labels chosen from a coarse-grained taxonomy including the categories: person, location, organization, date and numeric expression. Then, we combined the sequences of unique labels into one single token of the same label. Finally, we extracted the semantic phrase table from the augmented corpora in the same way mentioned above. This extracted a SPT containing about 135M phrase pair entries, which is about 30% smaller than the lexical PT. The resulting SPT is used to map phrases between NE-annotated source-target pairs, similar to PT matching. In addition to the advantages that were explained in Chapter 3, SPTs offer two more benefits in MT evaluation scenario:

  1. Their smaller size has positive impact on system's efficiency, due to the considerable search space reduction.
  2. The use of SPTs represents a promising direction to bring semantic knowledge into MT technology starting from the evaluation scenario.

A categorization of our features in terms of source/target derivation and adequacy/fluency nature is reported in Table 5.1.

5.4.2 Dataset

Datasets with manual evaluation of MT output have been made available through a number of shared evaluation tasks. However, most of these

datasets are not specifically annotated for adequacy measurement purposes, and the available adequacy judgements are limited to few hundred sentences for some language pairs. Moreover, most datasets are created by comparing reference translations with MT systems' output, disregarding the input sentences. Such judgements are hence biased towards the reference. Furthermore, the inter-annotator agreement is often low [Callison-Burch et al., 2007]. In light of these limitations, most of the available datasets are *per se* neither fully suitable for adequacy evaluation methods based on supervised learning, nor to provide stable and meaningful results. To partially cope with these problems, our experiments have been carried out over two different datasets:

1. **16K:** *16.000* English-Spanish pairs, containing four MT systems output created in a controlled environment to guarantee the **quality** of the annotations, annotated by professional translators trained on the task and based on clearly defined guidelines about the interpretation of the quality scores [Specia et al., 2010]. Translators were given the source sentence in English and its translation into Spanish, as produced by each of the four MT systems, and the quality judgement were assigned following these 4 point scale:
   1: requires complete re-translation.
   2: a lot of post editing needed (but quicker than re-translation).
   3: a little post editing needed.
   4: fit for purpose.

2. **WMT07:** *703* English-Spanish pairs, containing MT system output with an **adequacy** judgement for each pair, annotated by volunteers given the reference translation. The five point scale for adequacy, in this dataset, indicates how much of the meaning expressed in the reference translation is also expressed in a hypothesis translation (MT

system output):
5 = All, 4 = Most, 3 = Much, 2 = Little, 1 = None.

The two datasets present complementary advantages and disadvantages. On the one hand, although it is not annotated to explicitly capture meaning-related aspects of MT output, the quality oriented dataset has the main advantage of being large enough for supervised approaches. Moreover, it should allow to check the effectiveness of our feature set in estimating adequacy as a latent aspect of the more general notion of MT output quality. On the other hand, the smaller dataset is less suitable for supervised learning, but represents an appropriate benchmark for MT adequacy evaluation.

5.4.3 Algorithms and Approaches

In order to learn models for classification and regression we used Support Vector Machine (SVM) algorithms, which proved to be effective for a variety of NLP applications. To combine different features at various levels, different implementations of SVM were used in our experiments, namely:

1. LIBSVM Chang & Lin [2011] for *classification.*
2. SVM-Light Joachims [1999c] for *regression.*

5.5 Results

5.5.1 Adequacy and quality prediction

To experiment with our CLTE-based evaluation method minimizing overfitting, we randomized each dataset 5 times (D1 to D5), and split them into 80% for training and 20% for testing. Using different feature sets, we then trained and tested various regression models over each of the five

Features	D1	D2	D3	D4	D5	AVG
F	0.2506	0.2578	0.2436	0.2527	0.2443	0.25
SSyn+Syn	0.4387	0.4114	0.3994	0.4114	0.3793	0.41
F+SSyn+Syn	0.4215	0.4398	0.4059	0.4464	0.4255	0.428
F+SSyn+Syn+DR	0.4668	0.4602	0.4386	0.4437	0.4454	**0.451**
F+SSyn+Syn+DR+PT	0.4724	0.4715	0.4852	0.5028	0.4653	**0.48**
F+SSyn+Syn+DR+PT+SPT	0.4967	0.4802	0.4688	0.4894	0.4887	**0.485**
BLEU						0.2268
NIST						0.1953
TER						0.1938
METEOR						0.2713
QE						0.4792

Table 5.2: Pearson's correlation coefficient between our SVM regression model and human quality annotation, over the 16K dataset. Correlation achieved by standard MT automatic metrics is also reported.

splits, and computed correlation coefficients between the CLTE model predictions and the human gold standard annotations ([1-4] for quality, and [1-5] for adequacy).

16K quality-based dataset

In Table 5.2 we compare the Pearson's correlation coefficient of our SVM regression models against the results reported in Specia et al. [2009], calculated with four common MT evaluation metrics with a single reference: BLEU, NIST, TER and Meteor. For the sake of comparison, we also report the average quality correlation (QE) obtained by Specia et al. [2009] over the same dataset.[1]

The results show that the integration of syntactic and semantic information in our adequacy-oriented model allows to achieve a correlation with

[1]We only show the average results reported in Specia et al. [2009], since the distributions of the 16K dataset is different from our randomized distribution.

human quality judgements that is always significantly higher ($p < 0.05$) than the correlation obtained by the MT evaluation metrics used for comparison. As expected a considerable improvement over surface features is achieved by the integration of syntactic information. A further increase, however, is brought by the complementary contribution of SPT (*recall-oriented*, due to the higher coverage of semantics-aware phrase tables with respect to lexical PTs), and DR matching features (*precision-oriented*, due to the syntactic constraints posed to matching text portions). Although they are meant to capture meaning-related aspects of MT output, our features allow to outperform the results obtained by the generic quality-oriented features used by Specia et al. [2009], which do not discriminate between adequacy and fluency.[2] When dependency relations and phrase tables (both lexical and semantics-aware) are used in combination, our scores also outperform the average QE score. Finally, looking at the different random splits of the same dataset (D1 to D5), our correlation scores remain substantially stable, proving the robustness of our approach not only for adequacy, but also for quality estimation.

WMT07 adequacy-based dataset

In Table 5.3 we compare our regression model, obtained in the same way previously described, against three commonly used MT evaluation metrics Callison-Burch et al. [2007].

Due to the smaller size of the WMT07 dataset, the results reported do not show the same consistency over the 5 randomized datasets (D1 to D5). However, they still prove the effectiveness of our method in predicting MT output adequacy. Overall, the correlation achieved with features that only look at the source and the target is not far from other automatic

[2] As reported in Specia et al. [2009], more than 50% (39 out of 74) of the features used is translation-independent (only source-derived features).

Features	D1	D2	D3	D4	D5	AVG
F	0.10	0.03	0.04	0.10	0.14	0.083
SSyn+Syn	0.299	0.351	0.1834	0.2962	0.2417	0.274
F+SSyn+Syn	0.2648	0.2870	0.4061	0.3601	0.1327	0.29
F+SSyn+Syn+DR	0.3196	0.4568	0.2860	0.5057	0.4066	**0.395**
F+SSyn+Syn+DR+PT	0.3254	0.4710	0.3921	0.4599	0.3501	**0.40**
F+SSyn+Syn+DR+PT+SPT	0.3487	0.4032	0.4803	0.4380	0.3929	**0.413**
BLEU						0.466
TER						0.437
METEOR						0.357

Table 5.3: Pearson's correlation coefficient between our SVM regression model and human adequacy annotation over the WMT07 set. As term of comparison, correlation achieved by standard MT automatic metrics is also reported.

evaluation metrics that rely on the use of reference translations. Compared with Meteor, the correlation with human judgements is even higher.

5.5.2 Multi-class classification

To further explore the potential of our CLTE-based MT evaluation method, we trained an SVM multi-class classifier to predict the exact adequacy and quality scores assigned by human judges. The evaluation was carried out measuring the accuracy of our models with 10-fold cross validation to minimize overfitting. As a baseline, we calculated the performance of the Majority Class (MjC) classifier proposed in Specia et al. [2011], which labels all examples with the most frequent class among all classes. The performance improvement over the result obtained by the MjC baseline (Δ) has been calculated to assess the contribution of different feature sets.

Features	10-fold acc.	Δ
F	42.16%	5.16
Syn+SSyn	46.61%	9.61
F+Syn+SSyn	47.10%	10.10
F+Syn+SSyn+DR	47.26%	10.26
F+Syn+SSyn+DR+PT	48.15%	11.15
F+Syn+SSyn+DR+PT+SPT	**48.74**%	11.74
MjC	37%	-

Table 5.4: Multi-class classification accuracy of the quality/adequacy scores over 16K quality-based dataset.

16K quality-based dataset

The accuracy results reported in Table 5.4 show that also in this testing condition, syntactic and semantic features improve over surface form ones. Besides that, we observe a steady improvement over the MjC baseline (from 5% to 12%). This demonstrates the effectiveness of our adequacy-based features to predict exact quality scores in a 4-point scale, although this is a more challenging and difficult task than regression and binary classification. Such improvement is even more interesting considering that Specia et al. [2009] reported discouraging results with multi-class classification to predict quality scores. Moreover, while they claimed that removing target-independent features (*i.e.* those only looking at the source text) significantly degrades their QE performance, we achieved good results without using any of these features.

WMT07 adequacy-based dataset

As we can observe in Table 5.5, all variations of adequacy estimation models significantly outperform the MjC baseline, with improvements ranging from 14% to 20%. Interestingly, although the dataset is small and the

Features	10-fold acc.	Δ
F	50.07%	14.07
Syn+SSyn	54.19%	18.19
F+Syn+SSyn	54.34%	18.34
F+Syn+SSyn+DR	56.47%	20.47
F+Syn+SSyn+DR+PT	56.61%	20.61
F+Syn+SSyn+DR+PT+SPT	**56.75**%	20.75
MjC	36%	-

Table 5.5: Multi-class classification accuracy of the quality/adequacy scores over WMT07 adequacy-based dataset.

number of classes is higher (5-point scale), the improvement and overall results are better than those obtained on the 16K dataset. Such result confirms our hypothesis that adequacy-based features extracted from both source and target perform better on a dataset explicitly annotated with adequacy judgements.

In addition, the improvement over the MjC baseline (Δ) of our best model is much higher (20%) than the one reported in Specia et al. [2011] on adequacy estimation (6%). We are aware that their results are calculated over a dataset for a different language pair (*i.e.* English-Arabic) which brings up more challenges. However, our smaller dataset (700 vs 2580 pairs) and the higher number of classes (5 vs 4) compensate to some extent the difficulty of dealing with English-Arabic pairs.

5.5.3 Recognizing "good" vs "bad" translations

Last but not least, we considered the traditional scenario for quality and confidence estimation, which is a binary classification of translations into "good" and "bad" or, from the meaning point of view, "adequate" and "inadequate". Adequacy-oriented binary classification has many potential applications in the translation industry, ranging from the design of con-

Features	10-fold acc.	Δ
F	65.85%	11.85
Syn+SSyn	69.59%	15.59
F+Syn+SSyn	70.89%	16.89
F+Syn+SSyn+DR	71.39%	17.39
F+Syn+SSyn+DR+PT	71.92%	17.92
F+Syn+SSyn+DR+PT+SPT	**72.21**%	18.21
MjC	54%	-

Table 5.6: Accuracy of the binary classification into "good" and "bad" over 16K quality-based dataset.

fidence estimation methods that reward meaning-preserving translations, to the optimization of the translation workflow. For instance, an "adequate" translation can be just post-edited in terms of fluency by a target language native speaker, without having any knowledge of the source language. On the other hand, an "inadequate" translation should be sent to a human translator or to another MT system, in order to reach acceptable adequacy. Effective automatic binary classification has an evident positive impact on such workflow.

16K quality-based dataset

We grouped the quality scores in the 4-point scale into two classes, where scores {1,2} are considered as "bad" or "inadequate", while {3,4} are taken as "good" or "adequate". We carried out learning and classification using different sets of features with 10-fold cross validation. We also compared our accuracy with the MjC baseline, and calculated the improvement of each model (Δ) against it.

The results reported in Table 5.6 demonstrate that the accuracy of our models is always significantly superior to the MjC baseline. Moreover, also in this case there is a steady improvement using syntactic and semantic

Features	10-fold acc.	Δ
F	83.24%	12.84
Syn+SSyn	83.67%	13.27
F+Syn+SSyn	84.31%	13.91
F+Syn+SSyn+DR	84.86%	14.46
F+Syn+SSyn+DR+PT	84.96%	14.56
F+Syn+SSyn+DR+PT+SPT	**85.20**%	14.80
MjC	70.4%	-

Table 5.7: Accuracy of the binary classification into "good" and "bad" over WMT07 adequacy-based dataset.

features over the results obtained by surface form features. Additionally, it is worth mentioning that the best model improvement over the baseline (Δ) is much higher (about 18%) than the improvement reported in Specia et al. [2009] over the same dataset (about 8%), considering the average score obtained with their data distribution. This confirms the effectiveness of our CLTE approach also in classifying "good" and "bad" translations.

WMT07 adequacy-based dataset

We mapped the 5-point scale adequacy scores into two classes, with {1,2,3} judgements assigned to the "inadequate" class, and {4,5} judgements assigned to the "adequate" class. The main motivation for this distribution was to separate the examples in a way that adequate translations are substantially acceptable, while inadequate translations present evident meaning discrepancies with the source.

The results reported in Table 5.7 show that the accuracy of the binary classifiers to distinguish between "adequate" and "inadequate" classes was significantly superior (up to about 15%) to the MjC baseline. We also notice that surface form features have a significant contribution to deal with the adequacy-oriented dataset, while the gain obtained using syntactic

and semantic features (2%) is lower than the improvement observed in the 16K dataset. This might be due to the more unbalanced distribution of the classes which: *i)* leads to a high baseline, and *ii)* together with the small size of the WMT07 dataset, makes supervised learning more challenging. Finally, the improvement of all models (Δ) over the MjC baseline is much higher than the gain reported in Specia et al. [2011] over their adequacy-oriented dataset (around 2%).

5.6 Summary

In the effort of integrating semantics into MT technology, we focused on automatic MT evaluation, investigating the potential of cross-lingual textual entailment for adequacy assessment. The underlying assumption is that MT output adequacy can be determined by verifying that an entailment relation holds from the source to the target, and vice-versa. Within such framework, this work makes two main contributions.

First, in contrast with most current metrics based on the comparison between automatic translations and multiple references, we avoid the bottleneck represented by the manual creation of such references. CLTE, in fact, allows to evaluate the quality of MT output by looking only at source/target pairs.

Second, beyond current approaches biased towards fluency or general quality judgements, we isolate the adequacy dimension of the problem, exploring the potential of adequacy-oriented features extracted from the observation of source and target. To achieve our objectives, we successfully extended previous CLTE methods with a variety of linguistically motivated features. Altogether, such features led to reliable judgements that show high correlation with human evaluation. Coherent results on different datasets and classification schemes demonstrate the effectiveness of the

approach and its potential for different applications [Mehdad et al., 2012b].

We plan to explore the integration of our model as an error criterion in SMT system training. Although efficiency issues were out of the scope of this thesis, a necessary condition towards integrating our method in SMT technology in the future, is to optimize it in terms of efficiency.

Chapter 6

Crowdsourcing for CLTE Dataset Creation

6.1 Introduction

As for other NLP applications, both in monolingual and cross-lingual TE, the availability of large quantities of annotated data is an enabling factor for system development and evaluation. However, until now, the scarcity of such data on one hand, and the costs of creating new datasets of reasonable size on the other, have represented a bottleneck for a steady advancement towards achieving the state-of-the-art performance.

In the last few years, monolingual TE corpora for English and other European languages have been created and distributed in the framework of several evaluation campaigns, including the RTE Challenge,[1] the Answer Validation Exercise at CLEF,[2] and the Textual Entailment task at EVALITA.[3] Despite the differences in the design of these tasks, all the released datasets were collected through similar procedures, always involving expensive manual work by expert annotators.

Additionally, in the data creation process, large amounts of hand-crafted

[1] http://www.nist.gov/tac/2011/RTE/
[2] http://nlp.uned.es/clef-qa/ave/
[3] http://www.evalita.it/2009/tasks/te

T-H pairs often have to be discarded in order to retain only those featuring full agreement, in terms of the assigned entailment judgements, among multiple annotators. The amount of discarded pairs is usually high, thus contributing to the incremental costs of creating textual entailment datasets.[4]

The issues related to the shortage of datasets and the high costs for their creation are more evident in the CLTE scenario, since:

i) The task is relatively new and there are no available datasets for the development/evaluation cycle of CLTE algorithms. Moreover, there are no terms of comparison for cross-lingual methods with the monolingual ones.

ii) The application of the standard methods adopted to build RTE pairs requires proficiency in multiple languages, which significantly increases the costs of the data creation process.

To address these issues, in this chapter we devise cost-effective methodologies to create cross-lingual textual entailment corpora. In particular, we focus on two different strategies:

1. Taking advantage of an already available monolingual corpus, by casting the problem as a translation one. The challenge consists in taking a publicly available RTE dataset of English T-H pairs (*i.e.* the PASCAL-RTE3 dataset[5]) and create its English-Spanish CLTE equivalent by translating the hypotheses into Spanish.

2. Generating aligned CLTE corpora for different language combinations from scratch, without considering the available monolingual datasets.

The following sections overview our methodologies and experiments, carried out for both scenarios.

[4]For instance, in the first five RTE Challenges, the average effort needed to create 1,000 pairs featuring full agreement among 3 annotators was around 2.5 person-months. Typically, around 25% of the original pairs had to be discarded during the process, due to low inter-annotator agreement [Bentivogli et al., 2009].

[5]Available at: http://www.nist.gov/tac/data/RTE/index.html

6.2 Crowdsourcing

The availability and the increasing popularity of crowdsourcing services have been considered as an interesting opportunity to meet the aforementioned needs and design criteria.

One of the most popular crowdsourcing services is Amazon Mechanical Turk (MTurk)[6], "*a crowdsourcing Internet marketplace that enables computer programmers (known as Requesters) to co-ordinate the use of human intelligence to perform tasks which computers are unable to do [...] The Requesters are able to pose tasks known as HITs (Human Intelligence Tasks) [...] Workers [also known as "turkers"] can then browse among existing tasks and complete them for a monetary payment set by the Requester. To place HITs, the requesting programs use an open Application Programming Interface [...] Requesters can ask that Workers fulfill Qualifications before engaging a task, and they can set up a test in order to verify the Qualification. They can also accept or reject the result sent by the Worker, which reflects on the Worker's reputation. Currently, workers can have an address anywhere in the world [...] Requesters, which are typically corporations, pay 10 percent over the price of successfully completed HITs to Amazon*".[7]

Crowdsourcing services have been recently used with success for a variety of NLP applications [Callison-Burch & Dredze, 2010]. Although MTurk is directly accessible only to US citizens, the CrowdFlower service[8] provides a crowdsourcing interface to MTurk for non-US citizens.

The main idea in crowdsourcing the creation of NLP resources is that the acquisition and annotation of large datasets, needed to train and evaluate NLP tools and applications, can be carried out in a cost-effective manner

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

[7] Taken from http://en.wikipedia.org/wiki/Amazon_Mechanical_Turk

[8] http://crowdflower.com/

by defining simple Human Intelligence Tasks (HITs) routed to a crowd of non-expert workers (aka "Turkers") hired through on-line marketplaces.

The design of data acquisition HITs has to take into account several factors, each having a considerable impact on the difficulty of instructing the workers, the quality and quantity of the collected data, the time and overall costs of the acquisition. In addition, a major distinction has to be made between jobs requiring data *annotation*, and those involving content *generation*. In the former case, Turkers are presented with the task of labelling input data referring to a fixed set of possible values (*e.g.* making a choice between multiple alternatives, or assigning numerical scores to rank the given data). In the latter case, Turkers are faced with creative tasks consisting in the production of textual material (*e.g.* writing a correct translation, or a summary of a given text).

Overall, the ease of controlling the quality of the acquired data depends on the nature of the job. For annotation jobs, quality control mechanisms can be easily set up by calculating Turkers' agreement, by applying voting schemes, or by adding hidden gold units to the data to be annotated.[9] In contrast, the quality of the results of content generation jobs is harder to assess, due to the fact that multiple valid results are acceptable (*e.g.* the same content can be expressed, translated, or summarized in different ways). In such situations the standard quality control mechanisms are not directly applicable, and the detection of errors requires either costly manual verification at the end of the acquisition process, or more complex and creative solutions integrating HITs for quality check.

As regards textual entailment, the first work exploring the use of crowdsourcing services for data *annotation* is described in Snow et al. [2008],

[9]Both MTurk and CrowdFlower provide means to check workers' reliability, and weed out untrusted ones without money waste. These include different types of qualification mechanisms, the possibility of giving work only to known trusted Turkers (only with MTurk), and the possibility of adding hidden gold standard units in the data to be annotated (offered as a built-in mechanism only by CrowdFlower).

which shows high agreement between non-expert annotations of the RTE-1 dataset and existing gold standard labels assigned by expert labellers.

Their approach involves qualitative analysis of the collected data only *a posteriori*, after manual removal of invalid and trivial generated hypotheses. In contrast, our approaches integrate quality control mechanisms at all stages of the data collection/annotation process, thus minimizing the recourse to experts to check the quality of the collected material.

6.3 RTE3-derived CLTE dataset

In Chapter 3, we proposed CLTE as a generic framework for modelling language variability at the cross-lingual level. Obviously, any effort towards this direction becomes ineffective in the absence of CLTE datasets, since it would have been impossible to develop, evaluate and improve the solutions. As the first step in this direction, taking advantage of the available RTE-3 dataset, we cast the problem as translating the hypotheses into Spanish, hiring non-expert workers through the CrowdFlower channel to MTurk. Having a CLTE dataset originated from the available RTE data can also provide a term of comparison between the cross-lingual models and RTE monolingual approaches.

The following subsections overview our methodology and data acquisition process, the successive approximations that led to the definition of our methodology, and the lessons learned at each step. In order to verify the feasibility of our methodology in fast and cheap data creation, all experiments were carried out under strict time (10 days) and cost ($100) limitations.

6.3.1 Methodology

Starting from the RTE3 Development set (800 English T-H pairs), our corpus creation process has been organized in sentence *translation-validation* cycles, defined as separate "jobs" routed to CrowdFower's workforce. At the first stage of each cycle, the original English hypotheses are used to create a *translation* job for collecting their Spanish equivalents. At the second stage, the collected translations are used to create a *validation* job, where multiple judges are asked to check the correctness of each translation, given the English source. Translated hypotheses that are positively evaluated by the majority of trustful validators (*i.e.* those judged correct with a confidence above 0.8) are retained, and directly stored in our CLTE corpus together with the corresponding English texts. The remaining ones are used to create a new translation job. The procedure is iterated until substantial agreement for each translated hypothesis is reached. As regards the first phase of the cycle, we defined our **translation HIT** as follows:

In this task you are asked to:

- *First, judge if the Spanish sentence is a correct translation of the English sentence. If the English sentence and its Spanish translation are blank (marked as -), you can skip this step.*
- *Then, translate the English sentence above the text box into Spanish.*

Please make sure that your translation is:

1. *Faithful to the original phrase in both meaning and style.*
2. *Grammatically correct.*
3. *Free of spelling errors and typos.*

Don't use any automatic (machine) translation tool! You can have a look at any on-line dictionary or reference for the meaning of a word.

This HIT asks workers to first check the quality of an English-Spanish translation (used as a gold unit), and then write the Spanish translation of a new English sentence. The quality check allows to collect accurate translations, by filtering out judgments made by workers missing more than 20% of the gold units.

As regards the second phase of the cycle, our **validation HIT** has been defined as follows:

Su tarea es verificar si la traducción dada de una frase del Inglés al espaol es correcta o no. La traducción es correcta si:

1. *El estilo y sentido de la frase son fieles a los de la original.*
2. *Es gramaticalmente correcta.*
3. *Carece de errores ortográficos y tipográficos.*

Nota: el uso de herramientas de traducción automática (máquina) no está permitido!

This HIT asks workers to take binary decisions (Yes/No) for a set of English-Spanish translations including gold units. The title and the description are written in Spanish in order to weed out untrusted workers (*i.e.* those speaking only English), and attract the attention of Spanish speakers.

In our experiments, both the translation and validation jobs have been defined in several ways, trying to explore different strategies to quickly collect reliable data in a cost effective way. Such cost reduction effort led to the following differences between our work and similar related approaches documented in literature [Callison-Burch, 2009; Snow et al., 2008]:

- Previous works built on redundancy of the collected translations (up to 5 for each source sentence), thus resulting in more costly jobs. For instance, adopting a redundancy-based approach to collect 5 translations per sentence at the cost of $0.01 each, and 5 validations per translation at the cost of $0.002 each, would result in $80 for 800 sentences.

  Assuming that the translation process is complex and expensive, our cycle-based technique builds on simple and cheap validation mechanisms that drastically reduce the amount of translations required. In our case, 1 translation per sentence at the cost of $0.01, and 5 validations per translation at the cost of $0.002 each, would result in $32 for 800 sentences, making a conservative assumption of up to 8 iterations with 50% wrong translations at each cycle (*i.e.* 800 sentences in the first cycle, 400 in the second, 200 in the third, etc.).

- Previous works, involving validation of the collected data, are based on ranking/voting mechanisms, where workers are asked to order a number of translations, or select the best one given the source. Our approach to validation is based on asking workers to take binary decisions over source-target pairs. This results in an easier, faster, and eventually cheaper task.

- Previous works did not use any specific method to qualify the workers' knowledge, apart from *post-hoc* agreement computation. Our approach systematically includes gold units to filter out untrusted workers during the process. As a result we pay only for qualified judgments.

6.3.2 Experiments and lessons learned

The overall methodology, and the definition of the HITs described in Section 6.3.1, are the result of successive approximations that took into ac-

count two correlated aspects: the quality of the collected translations, and the current limitations of the CrowdFlower service. On one side, simpler, cheaper, and faster jobs launched in the beginning of our experiments had to be refined to improve the quality of the retained translations. On the other side, *ad-hoc* solutions had to be found to cope with the limited quality control functionalities provided by CrowdFlower. In particular, the lack of regional qualifications of the workers,[10] and of any qualification tests mechanism (useful features of MTurk) raised the need of defining more controlled, but also more expensive jobs.

Table 6.1 and the rest of this section summarize the progress of our work in defining the methodology adopted, the main improvements experimented at each step, the overall costs, and the lessons learned.

Step 1: a naïve approach. Initially, translation/validation jobs were defined without using qualification mechanisms, giving permission to any worker to complete our HITs. In this phase, our goal was to estimate the trade-off between the required development time, the overall costs, and the quality of translations collected in the most naïve conditions.

As expected, the job accomplishment time was negligible, and the overall cost very low. More specifically, it took about 1 hour for translating the 800 hypotheses at the cost of $12, and less than 6 hours to obtain 5 validations per each translation at the same cost of $12.

Nevertheless, as revealed by further experiments with the introduction of gold units, the quality of the collected translations was poor. In particular, 61% of them should have been rejected, often due to gross mistakes. As an example, among the collected material several translations in languages other than English revealed a massive and defective use of on-line translation tools by untrusted workers, as also observed by Callison-Burch

[10]This service was added to CrowdFlower after conducting our priliminary experiments, based on our request.

[2009].

Step 2: reducing *validation* errors. A first improvement addressed the validation phase, where we introduced *gold units* as a mechanism to qualify the workers, and consequently prune the untrusted ones. To this aim, we launched the validation HIT described in Section 6.3.1, adding around 50 English-Spanish control pairs. The pairs (equally distributed into positive and negative samples) have been extracted from the collected data, and manually checked by a Spanish native speaker.

The positive effect of using gold units has been verified in two ways. First, we checked the quality of the translations collected in the first naïve translation job, by counting the number of rejections (61%) after running the improved validation job. Then, we manually checked the quality of the translations retained with the new job. A manual check on 20% of the retained translations was carried out by a Spanish native speaker, resulting in 97% Accuracy. The 3% errors encountered are equally divided into minor translation errors, and controversial (but substantially acceptable) cases due to regional Spanish variations.

The considerable quality improvement observed has been obtained with a 25% increase in the cost (less than $3). However, as regards the accomplishment time, adding the gold units to qualify workers led to a considerable increase in duration (about 4 days for the first iteration). This is mainly due to the high number of automatically rejected judgments, obtained from untrusted workers missing the gold units. Because of the discrepancy between trusted and untrusted judgments, we faced another limitation of the CrowdFlower service, which further delayed our experiments. Often, in fact, the rapid growth of untrusted judgments activates automatic pausing mechanisms, based on the assumption that gold units are not accurate. This, however, is a strong assumption which does not

Elapsed time	Cost	Focus	Lessons learned
1 day	$24	Approaching CrowdFlower, defining a naïve methodology	Need of qualification mechanism, task definition in Spanish.
7 days	$58	Improving validation	Qualification mechanisms (gold units and regional) are effective, need of payment increase to boost speed.
9 days	$99.75	Improving translation	Combined HIT for qualification, payment increase worked!
10 days	$99.75	Obtaining bi-lingual RTE corpus	Fast, cheap, and reliable method.

Table 6.1: Creating a RTE3-derived CLTE dataset with $100 for a 10-day rush (summary and lessons learned). The reported costs are cumulative.

take into account the huge amount of non-qualified workers accepting (or even just playing with) the HITs.[11] For instance, in our case the vast majority of errors came from workers located in specific regions where the native language is not Spanish nor English.

Step 3: reducing *translation* errors. The observed improvement obtained by introducing gold units in the validation phase, led us to the definition of a new translation task, also involving a similar qualification mechanism. To this aim, due to language variability, it was clearly impossible to use reference translations as gold units. Taking into account the limitations of the CrowdFlower interface, which does not allow to set qualification tests or split the jobs into sequential subtasks (other effective and widely used features of MTurk), we solved the problem by defining the translation HITs as described in Section 6.3.1. This solution combines a validity check and a translation task, and proved to be effective with a decrease in the

[11]The auto-pausing system was modified by CrowdFlower after reporting the problems encountered.

translations eventually rejected (45%).

Step 4: reducing *time*. Considering the extra time required by using gold units, we decided to spend more money on each HIT to boost the speed of our jobs. In addition, to overcome the delays caused by the automatic pausing mechanism, we obtained from CrowdFlower the possibility to pose regional qualification, as commonly used in MTurk.

As expected, both solutions proved to be effective, and contributed to the final definition of our methodology. On one side, doubling the payment for each task (from \$0.01 to \$0.02 for each translation and from from \$0.002 to \$0.005 for each validation), we halved the required time to finish each job. On the other side, by imposing the regional qualification, we eventually avoided unexpected automatic pauses.

6.3.3 Results

The limited costs, together with the short time required to acquire reliable results, demonstrate the effectiveness of crowdsourcing services for simple sentence translation tasks. As a result, less than \$100 were spent in 10 days to define such methodology, leading to collect 426 pairs as a by-product. However, it's worth remarking that applying this technique to create the full corpus would cost about \$30. Following this successful methodology, we then launched the same HITs to translate all the hypotheses of the RTE-3 test set and the remaining hypotheses of RTE-3 development set, into Spanish. This resulted in the RTE3-derived CLTE dataset containing 1600 pairs (800 for training and 800 for test), which was released and used for our experiments in Chapter 3.

6.4 Content Synchronization CLTE dataset

In this section we devise another cost-effective methodology to create a cross-lingual textual entailment corpus from scratch. In particular, we concentrate our efforts on the following problems:

1. Is it possible to collect T-H pairs minimizing the intervention of expert annotators? To address this question, we explore the feasibility of crowdsourcing the corpus creation process. As a contribution beyond the few works on TE/CLTE data acquisition [Wang & Callison-Burch, 2010; Negri & Mehdad, 2010], we define an effective methodology that: *i)* does not involve experts in the most complex (and costly) stages of the process, *ii)* does not require pre-processing tools, and *iii)* does not rely on the availability of already annotated RTE corpora.

2. How can we guarantee good quality of the collected data at a low cost? We address the quality control issue through the decomposition of a complex task (*i.e.* creating and annotating entailment pairs) into smaller sub-tasks. Complex tasks are usually hard to explain in a simple way understandable to non-experts, difficult to accomplish, and not suitable for the application of the quality-check mechanisms provided by current crowdsourcing services. Our "divide and conquer" solution represents the first attempt to address a complex task involving content *generation* and *labelling* through the definition of a cheap and reliable pipeline of simple tasks which are easy to define, accomplish, and control.

3. Can we adapt such methodology to collect cross-lingual T-H pairs? We tackle this question by separating the problem of creating and annotating TE pairs from the issues related to the multilingual dimension. Our solution builds on the assumption that entailment an-

notations can be projected across aligned T-H pairs in different languages. In this case, a complex multilingual task is reduced to a sequence of simpler subtasks where the most difficult one, the generation of entailment pairs, is entirely monolingual. Besides ensuring cost-effectiveness, our solution allows us to overcome the problem of finding workers that are proficient in multiple languages. Moreover, since the core monolingual tasks of the process are carried out by manipulating English texts, we are able to address the very large community of English speaking workers, with a considerable reduction of costs and execution time.

Finally, as a by-product of our method, the acquired pairs are fully aligned for all language combinations, thus enabling meaningful comparisons between scenarios of different complexity (monolingual TE, and CLTE between close or distant languages).

Positioning our methodology among previous works, most of the approaches to content generation proposed so far rely on *post hoc* verification to filter out undesired low-quality data Mrozinski et al. [2008]; Mihalcea & Strapparava [2009]; Wang & Callison-Burch [2010]. The few solutions integrating validation HITs address the translation of single sentences [Bloodgood & Callison-Burch, 2010]. Compared to sentence translation, the task of creating CLTE pairs is both harder to explain without recurring to notions that are difficult to understand to non-experts (*e.g.* "semantic equivalence", "unidirectional entailment"), and harder to execute without mastering these notions.

To tackle these issues the "divide and conquer" approach described in the next section consists in the decomposition of a difficult *content generation* job into easier subtasks that are: *i)* self-contained and easy to explain, *ii)* easy to execute without any NLP expertise, and *iii)* suitable for the integration of a variety of runtime control mechanisms (regional

qualifications, gold units, "validation HITs") able to ensure a good quality of the collected material.

6.4.1 Methodology

Our approach builds on a pipeline of HITs routed to MTurk's workforce through the CrowdFlower interface. The objective is to collect aligned T-H pairs for different language combinations, reproducing an RTE-like annotation style. However, our annotation is not limited to the standard RTE framework, where only unidirectional entailment from T to H is considered. As a useful extension, we annotate any possible entailment relation between the two text fragments, including: *i)* bidirectional entailment (*i.e.* semantic equivalence between T and H), *ii)* unidirectional entailment from T to H, and *iii)* unidirectional entailment from H to T. The resulting pairs can be easily used to generate not only standard RTE datasets,[12] but also general-purpose collections featuring multi-directional entailment relations.

Data Acquisition and Annotation

We collect large amounts of CLTE pairs carrying out the most difficult part of the process (the creation of entailment-annotated pairs) at a monolingual level. Starting from a set of parallel sentences in n languages, (*e.g.* L1, L2, L3), n entailment corpora are created: *one* monolingual (L1/L1), and *n-1* cross-lingual (L1/L2, and L1/L3).

The monolingual corpus is obtained by modifying the sentences only in one language (L1). Original and modified sentences are then paired and annotated to form an entailment dataset for L1. The CLTE corpora are obtained by combining the modified sentences in L1 with the original sentences in L2 and L3, and projecting to the multilingual pairs the

[12] With the positive examples drawn from bidirectional and unidirectional entailments from T to H, and the negative ones drawn from unidirectional entailments from H to T.

annotations assigned to the monolingual pairs.

In principle, only two stages of the process require crowdsourcing multilingual tasks, but do not concern entailment annotations. The first one, at the beginning of the process, aims to obtain a set of parallel sentences to start with, and can be done in different ways (*e.g.* crowdsourcing the translation of a set of sentences). The second one, at the end of the process, consists of translating the modified L1 sentences into other languages (*e.g.* L2) in order to extend the corpus to cover new language combinations (*e.g.* L2/L2, L2/L3).

The execution of the two "multilingual" stages is not strictly necessary but depends on: *i)* the availability of parallel sentences to start the process, and *ii)* the actual objectives in terms of language combinations to be covered.[13]

As regards the first stage, in this work we started from a set of 467 English/Italian/German aligned sentences extracted from parallel documents downloaded from the Cafebabel European Magazine.[14] Concerning the second multilingual stage, we performed only one round of translations from English to Italian to extend the *3* combinations obtained without translations (ENG/ENG, ENG/ITA, and ENG/GER) with the new language combinations ITA/ITA, ITA/ENG, and ITA/GER.

The main steps of our corpus creation process, depicted in Figure 6.1, can be summarized as follows:

Step1: Sentence modification. The original English sentences (ENG) are modified through (monolingual) *generation* HITs asking Turkers to: *i)* preserve the meaning of the original sentences using different surface forms,

[13]Starting from parallel sentences in n languages, the n corpora obtained without recurring to translations can be augmented, by means of translation HITs, to create the full set of language combinations. Each round of translation adds 1 monolingual corpus, and *n-1* CLTE corpora.

[14]http://www.cafebabel.com/

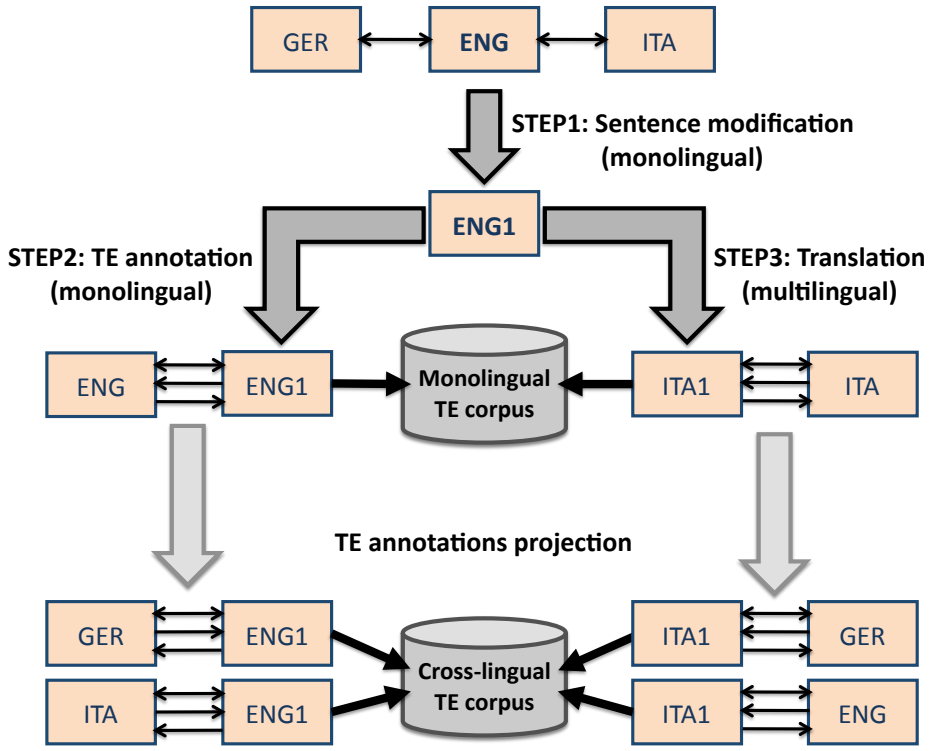

Figure 6.1: Content Synchronization corpus creation process.

or *ii)* slightly change their meaning by adding or removing content. Our assumption, in line with Bos et al. [2009], is that another way to think about entailment is to consider whether one text *T1* adds new information to the content of another text T: if so, then T is entailed by *T1*.

The result of this phase is a set of texts (ENG1) that can be of three types:

1. Paraphrases of the original ENG texts, that will be used to create bidirectional entailment pairs (ENG↔ENG1);

2. More specific sentences (the outcome of content addition operations), used to create ENG←ENG1 unidirectional entailment pairs;

3. More general sentences (the outcome of content removal operations), used to create ENG→ENG1 unidirectional entailment pairs.

Step2: TE Annotation. Entailment pairs composed of the original sentences (ENG) and the modified ones (ENG1) are used as input of (monolingual) *annotation* HITs asking Turkers to decide which of the two texts contains more information. As a result, each ENG/ENG1 pair is annotated as an example of unidirectional/bidirectional entailment, and stored in the monolingual English corpus. Since the original ENG texts are aligned with the ITA and GER texts, the entailment annotations of ENG/ENG1 pairs can be projected to the other language pairs and the ITA/ENG1 and GER/ENG1 pairs are stored in the CLTE corpus. The possibility of projecting TE annotations is based on the assumption that the semantic information is mostly preserved during the translation process. This particularly holds at the denotative level (*i.e.* regarding the truth values of the sentence) which is crucial to semantic inference. At other levels (*e.g.* lexical) there might be slight semantic variations which, however, are very unlikely to play a crucial role in determining entailment relations.

Step3: Translation. The modified sentences (ENG1) are translated into Italian (ITA1) through (multilingual) *generation* HITs reproducing the approach described in the previous section. As a result, three new datasets are produced by automatically projecting annotations: the monolingual ITA/ITA1, and the cross-lingual ENG/ITA1 and GER/ITA1.

Since the solution adopted for sentence translation does not present novelty factors, the remainder of this section will omit further details on it. Instead, the following sections will focus on the more challenging tasks of sentence modification and TE annotation.

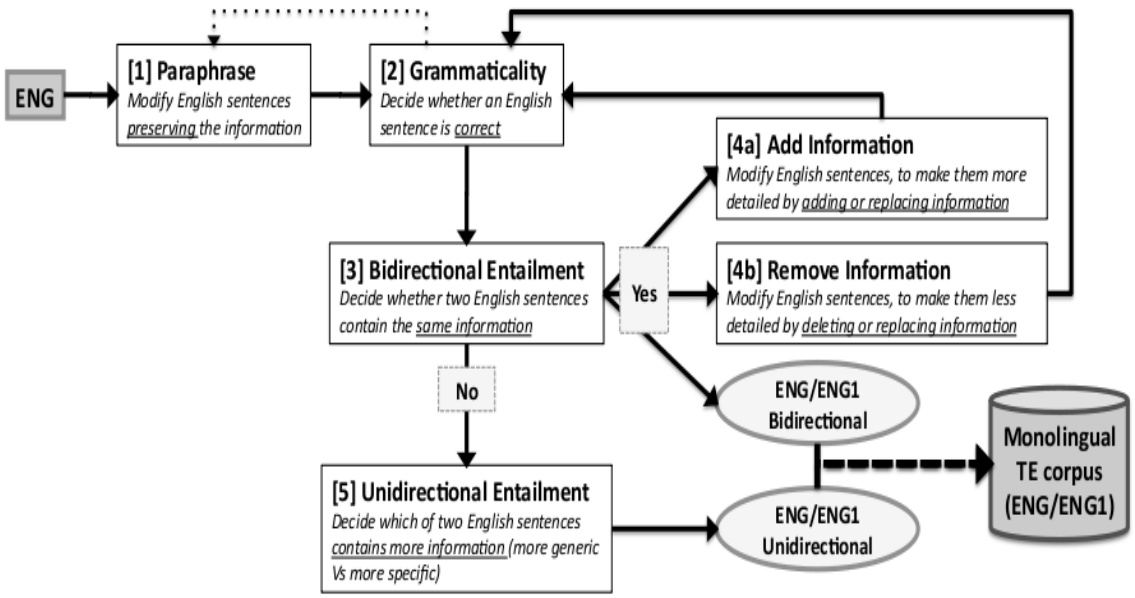

Figure 6.2: Sentence modification and TE annotation pipeline.

Crowdsourcing Sentence Modification and TE Annotation

Sentence modification and TE annotation have been decomposed into a pipeline of simpler monolingual English sub-tasks. Such pipeline, depicted in Figure 6.2, involves several types of generation/annotation HITs designed to be easily understandable to non-experts. Each HIT consists of: *i)* a set of instructions for a specific task (*e.g.* paraphrasing a text), *ii)* the data to be manipulated (*e.g.* an English sentence), and *iii)* a test to check workers' reliability. To cope with the quality control issues discussed in Section 6.2, such tests are realized using gold standard units, either hidden in the data to be annotated (annotation HITs) or defined as test questions that workers must correctly answer (generation HITs). Moreover, regional qualifications are applied to all HITs. As a further quality check, all the annotation HITs consider Turkers' agreement as a way to filter out low quality results (only annotations featuring agreement among 4 out of 5 workers are retained). The six HITs defined for each

subtask can be described as follows:

1. Paraphrase (generation). Modify an English text (ENG), in order to produce a semantically equivalent variant (ENG1). As a reliability test, before creating the paraphrase workers are asked to judge if two English sentences contain the same information.

2. Grammaticality (annotation). Decide if an English sentence is grammatically correct. This validation HIT represents a quality check of the output of each generation task (*i.e.* paraphrasing, and add/remove information HITs).

3. Bidirectional Entailment (annotation). Decide whether two English sentences, the original ENG and the modified ENG1, contain the same information (*i.e.* are semantically equivalent).

4a. Add Information (generation). Modify an English text to create a more specific one by adding content. As a reliability test, before generating the new sentence workers are asked to judge which of two given English sentences is more detailed.

4b. Remove Information (generation). Modify an English text to create a more general one by removing part of its content. As a reliability test, before generating the new sentence workers are asked to judge which of two given English sentences is less detailed.

5. Unidirectional Entailment (annotation). Decide which of two English sentences (the original ENG, and a modified ENG1) provides more information.

These HITs are combined in an iterative process that alternates text generation, grammaticality check, and entailment annotation steps. As a result, for each original ENG text we obtain multiple ENG1 variants of the three types (paraphrases, more general texts, and more specific texts) and, in turn, a set of annotated monolingual (ENG/ENG1) TE pairs.

As described in Section 6.4.1 (data acquisition and annotation), the resulting monolingual English TE corpus (ENG/ENG1) is used to create the following mono/cross-lingual TE corpora:

- ITA/ENG1, and GER/ENG1 (by projecting TE annotations)
- ITA/ITA1, GER/ITA1, and ENG/ITA1 (by translating the ENG1 texts into Italian, and projecting TE annotations)

6.4.2 Further Analysis

This section provides a quantitative and qualitative analysis of the results of our corpus creation methodology, focusing on the collected ENG-ENG1 monolingual dataset. It has to be remarked that, as an effect of the adopted methodology, all the observations and the conclusions drawn hold for the collected CLTE corpora as well.

Quantitative Analysis

Table 6.2 provides some details about each step of the pipeline. For each HIT the table presents: *i)* the number of items (sentences, or pairs of sentences) given in input, *ii)* the number of items (sentences or annotations) produced as output, *iii)* the number of items discarded when the agreement threshold was not reached, *iv)* the number of entailment pairs added to the corpus, *v)* the time (days and hours) required by the MTurk workforce to complete the job, and *vi)* the cost of the job.

In **HIT-1** (Paraphrase) 1,414 paraphrases were collected asking three different meaning-preserving modifications of each of the 467 original sentences.[15] From a practical point of view, such redundancy aims to ensure a sufficient number of grammatically correct and semantically equivalent

[15]Often, crowdsourced jobs return a number of output items that is slightly larger than required, due to the labour distribution mechanism internal to MTurk.

HIT	# Input	# Output	# Discarded	# Pairs	MTurk time	Cost ($)
1. Paraphrase	467	1,414			5d+10.5h	45.48
2. Grammaticality	1,414	1,326	88 (6.22%)		1d+15h	56.88
3. Bidirectional Ent.	1,326	1,213 (yes=1,205 no=8)	113 (8.52%)	301	3d+2h	53.47
4a. Add Info	452	916			3d	37.02
4b. Remove Info	452	923			2d+22h	29.73
2. Grammaticality	1,839	1,749	90 (4.89%)		2d+5h	64.37
3. Bidirectional Ent.	1,749	1,438 (yes=148 no=1,290)	311 (17.78%)	148	3d+20.5h	70.52
5. Unidirectional Ent.	1,298	1,171	127 (9.78%)	1,171 (491 + 680)	8.5h	78.24
TOTAL			721	**1,620**	**22d+11h**	**435.71**

Table 6.2: The monolingual dataset creation pipeline.

modified sentences. From a theoretical point of view, collecting many variants of a small pool of original sentences aims to create pairs featuring different entailment relations with similar superficial forms. This, in principle, should allow to obtain a dataset which requires TE systems to focus more on deeper semantic phenomena than on the surface realization of the pairs.

The collected paraphrases were sent as input to **HIT-2** (Grammaticality). After this validation HIT, the number of acceptable paraphrases was reduced to 1,326 (with 88 discarded sentences, corresponding to 6.22% of the total).

The retained paraphrases were paired with their corresponding original sentences, and sent to **HIT-3** (Bidirectional Entailment) to be judged for semantic equivalence. The pairs marked as bidirectional entailments (1,205) were divided in three groups: 25% of the pairs (301) were directly stored in the final corpus, while the ENG1 paraphrases of the remaining 75% (904) were equally distributed to the next modification steps.

In both **HIT-4a** (Add Information) and **HIT-4b** (Remove information) two new modified sentences were asked for each of the 452 paraphrases received as input. The sentences collected in these generation tasks were respectively 916 and 923.

The new modified sentences were sent back to **HIT-2** (Grammaticality) and **HIT-3** (Bidirectional Entailment). As a result 1,438 new pairs were created; out of these, 148 resulted to be bidirectional entailments and were stored in the corpus.

Finally, the 1,298 entailment pairs judged as non-bidirectional in the two previously completed HIT-3 (8+1,290) were given as input to **HIT-5** (Unidirectional Entailment). The pairs which passed the agreement threshold were classified according to the judgement received, and stored in the corpus as unidirectional entailment pairs.

The analysis of Table 6.2 allows to formulate some considerations. First, the percentage of discarded items confirms the effectiveness of decomposing complex generation tasks into simpler subtasks that integrate validation HITs and quality checks based on non-experts' agreement. In fact, on average, around 9.5% of the generated items were discarded without experts' intervention.[16] Second, the amount of discarded items gives evidence about the relative difficulty of each HIT. As expected, we observe lower rejection rates, corresponding to higher inter-annotator agreement, for grammaticality HITs (5.55% on average) than for more complex entailment-related tasks (12.02% on average).

Looking at costs and execution time, it is hard to draw definite conclusions due to several factors that influence the progress of the crowdsourced jobs (*e.g.* the fluctuations of Turkers' performances, the time of the day at which jobs are posted, the difficulty to set the optimal cost for a given HIT).[17] On the one hand, as expected, the more creative "Add Info" task proved to be more demanding than the "Remove Info": even though it

[16]Moreover, it is worthwhile noticing that around 20% of the collected items were automatically rejected (and not paid) due to failures on the gold standard controls created both for generation and annotation tasks.

[17]The payment for each HIT was set on the basis of a previous feasibility study aimed at determining the best trade-off between cost and execution time. However, replicating our approach would not necessarily result in the same costs.

was paid more, it still took little more time to be completed. On the other hand, although the "Unidirectional Entailment" task was expected to be more difficult and thus rewarded more than the "Bidirectional Entailment" one, in the end it took notably less time to be completed. Nevertheless, the overall figures (435 $, and about 22.5 days of MTurk work to complete the process)[18] clearly demonstrate the effectiveness of the approach. Even considering the time needed for an expert to manage the pipeline (*i.e.* one week to prepare gold units, and to handle the I/O of each HIT), these figures show that our methodology provides a cheaper and faster way to collect entailment data in comparison with the RTE average costs reported in Section 6.1.

As regards the amount of data collected, the resulting corpus contains 1,620 pairs with the following distribution of entailment relations: *i)* 449 bidirectional entailments, *ii)* 491 ENG→ENG1 unidirectional entailments, and *iii)* 680 ENG←ENG1 unidirectional entailments.

It must be noted that our methodology does not lead to the creation of pairs where some information is provided in one text and not in the other, and vice-versa, as Example 1 shows:

Example 1.

ENG: New theories were emerging in the field of psychology.

ENG1: New theories were rising, which announced a kind of veiled racism.

These negative examples in both directions represent a natural extension of the dataset, relevant also for specific application-oriented scenarios, and their creation will be addressed in future work.

Besides the achievement of our primary objectives, the adopted approach led to some interesting by-products. First, the generated corpora

[18] Although by projecting annotations the ENG1/ITA and ENG1/GER CLTE corpora came for free, the ITA1/ITA, ITA1/ENG, and ITA1/GER combinations created by crowdsourcing translations added 45 USD and approximately 5 days to these figures.

are perfectly suitable to produce entailment datasets similar to those used in the traditional RTE evaluation framework. In particular, considering any possible entailment relation between two text fragments, our annotation subsumes the one proposed in RTE campaigns. This allows for the cost-effective generation of RTE-like annotations from the acquired corpora by combining ENG↔ENG1 and ENG→ENG1 pairs to form 940 positive examples (449+491), keeping the 680 ENG←ENG1 as negative examples. Moreover, by swapping ENG and ENG1 in the unidirectional entailment pairs, 491 additional negative examples and 680 positive examples can be easily obtained.

Finally, the output of HITs 1-2-3 in Table 6.2 represents *per se* a valuable collection of 1,205 paraphrases. This suggests the great potential of crowdsourcing for paraphrase acquisition.

Qualitative Analysis

Through manual verification of more than 50% of the corpus (900 pairs), a total number of 53 pairs (5.9%) were found incorrect. The different errors were classified as follows:

Type 1: Sentence modification errors. Generation HITs are a minor source of errors, being responsible for 10 problematic pairs. These errors are either introduced by generating a false statement (Example 2), or by forming a not fully understandable, awkward, or non-natural sentence (Example 3).

Example 2.

ENG: Kosovo was the subject of major riots in 1989.
ENG1: The Russian city of Kosovo was the subject of ...

Example 3.

ENG: Balat is the Kurdish-Armenian district of Instanbul.
ENG1: Balat is a place, which is the Kurdish-Armenian ...

Type 2: TE annotation errors. The notion of containing more/less information, used in the "Unidirectional Entailment" HIT, can mostly be applied straightforwardly to the entailment definition. However, the concept of "more/less detailed", which generally works for factual statements, in some cases is not applicable. In fact, the MTurk workers have regularly interpreted the instructions about the amount of information as concerning the quantity of concepts contained in a sentence. This is not always corresponding to the actual entailment relation between the sentences. As a consequence, 43 pairs featuring wrong entailment annotations were encountered. These errors can be classified as follows:

a) 13 pairs, where the added/removed information changes the meaning of the sentence. In these cases, the modified sentence was judged more/less specific than the original one, leading to unidirectional entailment annotation. On the contrary, in terms of the standard entailment definition, the correct annotation is "no entailment" (as in Example 4, which was annotated as ENG→ENG1):

Example 4.

ENG: If you decide to live in Bulgaria, you have to like difficulties because they are not difficulties, they are challenges.
ENG1: You have to like difficulties as they are not difficulties, they are challenges.

b) 10 pairs where the incorrect annotation is due to a coreference problem, as in:

Example 5.

ENG: John Smith is the new CEO of the company.
ENG1: He is the new CEO of the company.

These pairs were labelled as unidirectional entailments (in the example above ENG→ENG1), under the assumption that a proper name is more specific and informative than a pronoun. However, adhering to the TE definition, co-referring expressions are equivalent, and their realization does not play any role in the entailment decision. This implies that the correct entailment annotation is "bidirectional".

c) 9 pairs where the sentences are semantically equivalent, but contain a piece of information which is explicit in one sentence, and implicit in the other. In these cases, Turkers judged the sentence containing the explicit mention as more specific, and thus the pair was annotated as unidirectional entailment.

Example 6.

ENG: I hear the click of the trigger and the burst of bullets reach me immediately.
ENG1: I hear the trigger and the burst of bullets reach me instantly.

In Example 6, the expression *"the trigger"* in ENG1 implicitly means *"the click of the trigger"*, making the two sentences equivalent, and the entailment bidirectional (instead of ENG→ENG1).

d) 7 pairs where the information removed from or added to the sentence is not relevant to the entailment relation. In these cases, the modified sentence was judged less/more specific than the original one (and thus considered as unidirectional entailment), even though the correct judgement is "bidirectional", as in:

Example 7.

ENG: At the same time, AKP is struggling with its approach to the EU.
ENG1: AKP is struggling with its approach to the European Union.

e) 4 pairs where the added/removed information concerns universally quantified general statements, about which the interpretation of "more/less specific" given by Turkers resulted in the wrong annotation.

Example 8.

ENG: I think the success of multicultural couples depends on the size of the cultural gap between the two partners
ENG1: I believe the success of the couples depends on the size of the cultural gap between the 2 partners.

In Example 8, the additional information (*"multicultural"*) restricts the set to which it refers (*"couples"*) making ENG entailed by ENG1, and not vice versa as resulted from Turkers' annotation.

In light of this analysis, we conclude that the sentence modification methodology proved to be successful, as the low number of Type 1 errors shows. Considering that the most expensive phase in the creation of a TE dataset is the generation of the pairs, this is a significant achievement. Differently, the entailment assessment phase appears to be more problematic, accounting for the majority of errors. As shown by Type 2 errors, this is due to a partial misalignment between the instructions given in our HITs, and the formal definition of textual entailment. For this reason, further experimentation will explore different ways to instruct workers (*e.g.* asking to consider proper names and pronouns as equivalent) in order to reduce the amount of errors produced. As a final remark, considering that in the creation of a TE dataset the manual check of the annotated pairs represents a minor cost, even the involvement of experts to filter out wrong annotations would not decrease the cost-effectiveness of the proposed methodology.

Results

The result of our work is the first large-scale dataset containing more than 1,600 aligned pairs for several combinations of texts-hypotheses in English, Italian, and German. Among the advantages of our method it is worth mentioning: *i)* the full alignment between the created corpora, *ii)*

the possibility to easily extend the dataset to new languages, and *iii)* the feasibility of creating general-purpose corpora, featuring multi-directional entailment relations, that subsume the traditional RTE-like annotation.

In order to take advantage of such dataset, 800 pairs were manually checked. Then, to provide a more challenging scenario, balancing the entailment judgements and decreasing a correlation with the length of the sentences, 500 balanced pairs for each language pair were extracted. This dataset has been used for content synchronization application experiments in Chapter 4.

It is worth to mention that the resulting dataset is made freely available for research purposes through the website of the funding EU Project CoSyne,[19] to contribute in meeting the strong need for resources to develop and evaluate novel applications for textual entailment.

6.5 Summary

There is an increasing need of annotated data to develop new solutions to the TE problem, explore new entailment-related tasks, and set up experimental frameworks targeting real-world applications.

As a first step in this direction, we took advantage of an already existing monolingual English RTE corpus, casting the problem as a translation task where Spanish translations of the hypotheses are collected and validated by the workers. As a result, we collected the first CLTE datasets, containing 1600 entailment pair for English-Spanish aligned with the original monolingual RTE-3 dataset.

In light of this positive experience, in the next step, we explored crowdsourcing data acquisition methods to address the complementary problem of collecting new cross-lingual entailment pairs from scratch. Besides

[19] http://www.cosyne.eu/

that, we considered cost effectiveness and replicability as additional requirements. To achieve our objectives, we developed a "divide and conquer" methodology based on crowdsourcing. Our approach presents several key innovations with respect to the related works on TE data acquisition. These include the decomposition of a complex content generation task in a pipeline of simpler subtasks accessible to a large crowd of non-experts, and the integration of quality control mechanisms at each stage of the process.

The result of our work created the first large-scale dataset containing both monolingual and cross-lingual corpora for several combinations of texts-hypotheses in English, Italian, and German. Among the advantages of our method it is worth mentioning: *i)* the full alignment between the created corpora, *ii)* the possibility to easily extend the dataset to new languages, and *iii)* the feasibility of creating general-purpose corpora, featuring multi-directional entailment relations, that subsume the traditional RTE-like annotation. We used the created datasets to develop and improve CLTE algorithms (details are available in Chapter 3) and its application in content synchronization scenario (details are available in Chapter 4).

Chapter 7

Conclusion

7.1 Recapitulation

This thesis proposed and discussed Cross-Lingual Textual Entailment [Mehdad et al., 2010b], as a framework to cross the semantic and inference barriers across languages. Our work aims at providing models and insights, not only to bring together machine translation and textual entailment research, but also to deploy effective components for different application scenarios ranging from content synchronization to MT evaluation.

In this direction, taking advantage of our research in monolingual textual entailment [Mehdad et al., 2010a; Mehdad, 2009; Mehdad & Magnini, 2009a; Kouylekov et al., 2010a, 2011], in Mehdad et al. [2010b], we proposed a pivoting approach to CLTE. We took advantage of available MT components, using them at the front-end of existing TE engines. The motivation of this solution is that a modular pivoting architecture is easier to develop, debug, and maintain. Moreover, it allows for easy extensions to other languages by just adding extra MT systems (in terms of language pairs). Through different experiments over the two datasets, we achieved promising results, which are even more encouraging considering that, at the cross-lingual level, we obtained results comparable to those calculated over the original monolingual datasets.

While the pivoting approach has been promising, the availability of MT components and the noise introduced by translation errors are among the limitations of such method. To cope with these limitations, we took advantage of a tighter integration of MT and TE algorithms and techniques by proposing an integrated solution [Mehdad et al., 2011]. By extracting the lexical and semantic knowledge from parallel corpora, by using and extending TE techniques, we could avoid dependencies on external MT components. We also successfully extended our previously integrated method with syntactic and semantic features, that lead to the results which outperform those obtained with the pivoting solution.

To further support our claims about the usefulness of CLTE, and the effectiveness of the proposed solutions, we successfully applied them in two interesting application scenarios. Firstly, we have addressed the task of synchronizing the content of two documents about the same topic written in different languages. Using a combination of lexical, syntactic, and semantic features to create a CLTE system, we reported several experiments over different datasets proving the feasibility of detecting semantic equivalence and information disparity by means of CLTE [Mehdad & Negri, 2012; Mehdad et al., 2012a].

Secondly, we focused on automatic MT evaluation, investigating the potential of CLTE for adequacy assessment avoiding the use of reference translations. In this direction, we could isolate the adequacy dimension of the problem, exploring the potential of adequacy-oriented features extracted from the observation of source and target words. Our use of various sets of linguistically motivated features led to reliable judgements that show high correlation with human evaluation in different experimental setups. Moreover, promising results on different classification schemes demonstrated the effectiveness of our approach for integrating semantics into MT technology [Mehdad et al., 2012b].

Since CLTE was proposed as the core problem of this thesis for the first time, proving the success, effectiveness and potential of the approaches mentioned above could have not been possible without suitable CLTE datasets. To provide large quantities of annotated data to enable the system development phase (*e.g.* to tune cross-lingual models), we presented [Negri et al., 2011; Negri & Mehdad, 2010] cheap and fast and effective automatic procedures to create CLTE datasets by crowdsourcing. As a result we collected the first dataset containing both monolingual and cross-lingual corpora for several combinations of texts-hypotheses in English, Italian, and German.

In parallel with this work, a task called Cross-lingual Textual Entailment for Content Synchronization (CLTE@SemEval-2012, task #8),[1] has been organized within SemEval 2012 [Negri et al., 2012]. This initiative aims at promoting the research topics proposed on this thesis among the NLP community, and bring the semantics and MT communities closer. We believe that research in this direction can greatly benefit from MT-derived techniques and, at the same time, contribute to a variety of MT-related tasks, ranging from re-scoring MT outputs to adequacy evaluation. We also believe that content synchronization represents a challenging application scenario to test the capabilities of advanced NLP systems.

7.2 Future direction

Some of the issues addressed in this thesis raise interesting questions, problems and future research directions. In Chapter 2, we investigated possible solutions for monolingual TE including kernel based semantic/syntactic learning, phrasal matching algorithm, and extracting new lexical resources. On one side, the phrasal matching method allows us to use a large collection

[1] http://www.cs.york.ac.uk/semeval-2012/task8/

of paraphrases but limits the algorithm to the use of only lexical resources. One the other side, semantic/syntactic kernels have proved to be efficient and more accurate, but do not allow the use of paraphrases. Integrating those two solutions, by moving form token based to phrase based semantic/syntactic kernels, could open new interesting research directions.

In the context of chapter 3, overall results using different models suggest that adding relevant linguistic features can improve CLTE performance. These findings suggest that cross-lingual topic modeling and Wikipedia entity linking could also contribute in the advancement of such models and approaches.

Next, in the machine translation research community, there is a high interest in integrating MT technology within applications such as computer-aided translation tools. Recent European projects like "Machine Translation Enhanced Computer Assisted Translation (MateCat)", "Cognitive Analysis and Statistical Methods for Advanced Computer Aided Translation (CASMACAT)" and "Moses Open Source Evaluation and Support Coordination for OutReach and Exploitation (MOSESCORE)" confirm such interest. Further works for improving our adequacy evaluation method and integrating it into SMT for optimization purposes could be beneficial for these projects. On one hand, exploring new features capturing other semantic dimensions can be further investigated. On the other hand, exploring the integration of our method as an error criterion in SMT system training can be further studied. A prerequisite towards integrating our method in SMT technology at future is efficiency optimization.

In TE research community the use of machine learning methods has been always dominant. Obviously, training data are essential if the core method for approaching TE is supervised learning. Returning to our automatic content synchronization experiments in Chapter 5, one interesting direction that can be investigated further is to tackle issues related to "unknown"

cases, that are not covered by the available datasets. Moreover, we are interested in exploring the impact of having more training data for such application scenarios.

Last but not least, we believe that applications of CLTE are not limited to the ones discussed in this thesis. Indeed, it would be interesting to take advantage of the cross-lingual semantic and inference framework to deal with other multilingual application scenarios ranging from MT output re-ranking to multilingual semantic search and knowledge representation.

Bibliography

Agerri, R. (2008) Metaphor in Textual Entailment. In: Proceedings of COLING 2008, 22nd International Conference on Computational Linguistics, Posters Proceedings, 18-22 August 2008, Manchester, UK. 14

Agichtein, E., Askew, W. & Liu, Y. (2008) Combining Lexical, Syntactic, and Semantic Evidence for Textual Entailment Classification. Proceedings of TAC . 31

Aharon, R., Szpektor, I. & Dagan, I. (2010) Generating Entailment Rules from Framenet. In: Proceedings of the ACL 2010 Conference Short Papers. Association for Computational Linguistics, pp. 241–246. 20

Androutsopoulos, I. & Malakasiotis, P. (2010) A Survey of Paraphrasing and Textual Entailment Methods. Journal of Artificial Intelligence Research 38(1):135–187. 92

Avramidis, E., Popovic, M., Torres, D. V. & Burchardt, A. (2011) Evaluate with Confidence Estimation: Machine Ranking of Translation Outputs using Grammatical Features. In: Proceedings of the Sixth Workshop on Statistical Machine Translation. Workshop on Statistical Machine Translation (WMT-11), located at EMNLP, July 30-31, Edinburgh, United Kingdom. Association for Computational Linguistics. 123

Bach, N., Huang, F. & Al-Onaizan, Y. (2011) Goodness: A Method for

Measuring Machine Translation Confidence. In: 49th Annual Meeting of the Association for Computational Linguistics. pp. 211–219. 123

Baker, C. F., Fillmore, C. J. & Lowe, J. B. (1998) The Berkeley FrameNet Project. In: Proceedings of the 17th international conference on Computational linguistics - Volume 1. Association for Computational Linguistics, Stroudsburg, PA, USA, COLING '98. 19

Balahur, A., Lloret, E., Ferrández, O., Montoyo, A., Palomar, M. & Muñoz, R. (2008) The DLSIUAES Team's Participation in the TAC 2008 Tracks. In: Proceedings of the Text Analysis Conference (TAC). 19

Banerjee, S. & Lavie, A. (2005) METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In: Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization. pp. 65–72. 119

Bannard, C. & Callison-Burch, C. (2005) Paraphrasing with Bilingual Parallel Corpora. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 597–604. 25, 70, 86

Bar-Haim, R., Berant, J., Dagan, I., Greental, I., Mirkin, S., Shnarch, E. & Szpektor, I. (2008) Efficient Semantic Deduction and Approximate Matching over Compact Parse Forests. In: Proceedings of Text Analysis Conference (TAC). Gaithersburg, Maryland USA. 19, 28

Bensley, J. & Hickl, A. (2008) Application of LCC's GROUNDHOG System for RTE-4. In: Proceedings of the Fourth PASCAL Challenges Workshop on Recognizing Textual Entailment. Gaithersburg, Maryland, USA. 31

Bentivogli, L., Pianta, E. & Girardi, C. (2002) Multiwordnet: Developing an Aligned Multilingual Database. In: First International Conference on Global WordNet, Mysore, India. 71

Bentivogli, L., Dagan, I., Dang, H., Giampiccolo, D. & Magnini, B. (2009) The Fifth Pascal Recognizing Textual Entailment Challenge. In: Proceedings of TAC 2009 Workshop. 15, 18, 142

Bentivogli, L., Cabrio, E., Dagan, I., Giampiccolo, D., Leggio, M. & Magnini, B. (2010a) Building Textual Entailment Specialized Data Sets: a Methodology for Isolating Linguistic Phenomena Relevant to Inference. Proceedings of LREC 2010 . 17

Bentivogli, L., Clark, P., Dagan, I., T. Dang, H. & Giampiccolo, D. (2010b) The Sixth PASCAL Recognizing Textual Entailment Challenge. In: Proceedings of TAC'2010. 15, 16, 18, 70

Bentivogli, L., Clark, P., Dagan, I., T. Dang, H. & Giampiccolo, D. (2011) The Seventh PASCAL Recognizing Textual Entailment Challenge. In: Proceedings of TAC'2011. 15, 16, 18

Berant, J., Dagan, I. & Goldberger, J. (2011) Global Learning of Typed Entailment Rules. Proceedings of ACL, Portland, OR . 20

Bernard, M., Boyer, L., Habrard, A. & Sebban, M. (2008) Learning Probabilistic Models of Tree Edit Distance. Pattern Recognition 41(8):2611–2629. 33

Blatz, J., Fitzgerald, E., Foster, G., Gandrabur, S., Goutte, C., Kulesza, A., Sanchis, A. & Ueffing, N. (2004) Confidence Estimation for Machine Translation. In: Proceedings of the 20th international conference on Computational Linguistics. Association for Computational Linguistics, pp. 315–321. 122, 123

Bloehdorn, S. & Moschitti, A. (2007) Combined Syntactic and Semantic Kernels for Text Classification. In: ECIR. 53

Bloodgood, M. & Callison-Burch, C. (2010) Using Mechanical Turk to Build Machine Translation Evaluation Sets. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics, pp. 208–211. 154

Bogdan, S., Constantin, O., Christian, S., Shiyan, O., Oscar, F., Milen, K. & Matteo, N. (2008) Entailment-based Question Answering for Structured Data. In: Coling 2008: Companion volume: Posters and Demonstrations. Manchester, UK, pp. 29–32. 13

Bos, J. (2005) Combining Shallow and Deep NLP Methods for Recognizing Textual Entailment. In: Proceedings of the PASCAL RTE Challenge. pp. 65–68. 19

Bos, J. & Markert, K. (2005) Recognising Textual Entailment with Logical Inference. In: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 628–635. 20, 28

Bos, J. & Oka, T. (2007) A spoken Language Interface with a Mobile Robot. Artificial Life and Robotics 11(1):42–47. 14

Bos, J., Zanzotto, F. & Pennacchiotti, M. (2009) Textual Entailment at EVALITA 2009. Proceedings of EVALITA . 157

Burchardt, A., Reiter, N., Thater, S. & Frank, A. (2007) Semantic Approach to Textual Entailment: System Evaluation and Task Analysis. In: Proceedings of the 3rd-PASCAL Workshop on Textual Entailment. Prague. 49, 62

Burchardt, A., Pennacchiotti, M., Thater, S. & Pinkal, M. (2009) Assessing the Impact of Frame Semantics on Textual Entailment. Nat. Lang. Eng. 15:527–550. 19

Cabrio, E., Kouylekov, M. & Magnini, B. (2008) Combining Specialized Entailment Engines for RTE-4. In: Proceedings of TAC08, 4th PASCAL Challenges Workshop on Recognising Textual Entailment. 39

Callison-Burch, C. (2009) Fast, Cheap, and Creative: Evaluating Translation Quality Using Amazon's Mechanical Turk. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Singapore, pp. 286–295. 147, 149

Callison-Burch, C. & Dredze, M. (2010) Creating Speech and Language Data with Amazon's Mechanical Turk. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics, pp. 1–12. 143

Callison-Burch, C., Fordyce, C., Koehn, P., Monz, C. & Schroeder, J. (2007) (Meta-)Evaluation of Machine Translation. In: Proceedings of the Second Workshop on Statistical Machine Translation (WMT07). Association for Computational Linguistics. 129, 132

Callison-Burch, C., Koehn, P., Monz, C., Peterson, K., Przybocki, M. & Zaidan, O. (2010) Findings of the 2010 Joint Workshop on Statistical Machine Translation and Metrics for Machine Translation. In: Proceedings of the Joint Fifth Workshop on Statistical Machine Translation and MetricsMATR. Association for Computational Linguistics, pp. 17–53. 122

Carreras, X., Chao, I., Padró, L. & Padró, M. (2004) Freeling: An Open-Source Suite of Language Analyzers. In: Proceedings of the 4th LREC. vol. 4. 126

Chang, C. & Lin, C. (2011) LIBSVM: a Library for Support Vector Machines. ACM Transactions on Intelligent Systems and Technology (TIST) 2(3):27. 130

Charniak, E. (2000) A Maximum-Entropy-Inspired Parser. In: Proceedings of the 1st NAACL conference. 56

Chklovski, T. & Pantel, P. (2004) VerbOcean: Mining the Web for Fine-Grained Semantic Verb Relations. In: Proceedings of EMNLP 2004. Association for Computational Linguistics, Barcelona, Spain. 19

Cingolani, P. (2005) JSwarm-PSO: Particle Swarm Optimization Package. Available at http://jswarm-pso.sourceforge.net/. 37

Clark, P. & Harrison, P. (2009) An Inference-based Approach to Recognizing Entailment. In: Proceedings of TAC 2008. 18

Clinchant, S., Goutte, C. & Gaussier, E. (2006) Lexical Entailment for Information Retrieval. In: Proceedings of ECIR'06. 65

Collins, M. & Duffy, N. (2002) New Ranking Algorithms for Parsing and Tagging: Kernels over Discrete Structures, and the Voted Perceptron. In: Proceedings of ACL '02. 32, 52

Crammer, K. & Singer, Y. (2002) On the Algorithmic Implementation of Multiclass Kernel-based Vector Machines. The Journal of Machine Learning Research 2:265–292. 110

Cristianini, N. & Holloway, R. (2001) Latent Semantic Kernels. 54

Dagan, I. & Glickman, O. (2004) Probabilistic Textual Entailment: Generic Applied Modeling of Language Variability. In: Proceedings of the PASCAL Workshop of Learning Methods for Text Understanding and Mining. 1, 11, 14, 65, 100, 124

Dagan, I., Glickman, O. & Magnini, B. (2005) The Pascal Recognising Textual Entailment Challenge. In: Proceedings of the First PASCAL Challenges Workshop on Recognising Textual Entailment. 1, 11, 14, 15

De Marneffe, M., MacCartney, B. & Manning, C. (2006) Generating Typed Dependency Parses from Phrase Structure Parses. In: Proceedings of LREC. vol. 6, pp. 449–454. 107

Delmonte, R., Bristot, A., Aldo, M., Boniforti, P., Tonelli, S., Ca, U. & Bembo, F. C. (2007) Entailment and Anaphora Resolution in RTE-3. In: Proceedings of the ACL-07 Workshop on Textual Entailment and Paraphrasing. 19

Denkowski, M. & Lavie, A. (2010) Extending the METEOR Machine Translation Evaluation Metric to the Phrase Level. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 250–253. 25, 86

Dinu, G. & Wang, R. (2009) 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. Association for Computational Linguistics, pp. 211–219. 25, 86

Doddington, G. (2002) Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics. In: Proceedings of the second international conference on Human Language Technology Research. Morgan Kaufmann Publishers Inc., pp. 138–145. 119

Doina, T., Diana, M. A. & Dana, L. (2008) Text Entailment for Logical Segmentation and Summarization. In: NLDB '08: Proceedings of the 13th international conference on Natural Language and Information Systems. Springer-Verlag, Berlin, Heidelberg, pp. 233–244. 13

Dragomir, R. (2000) A Common Theory of Information Fusion From Multiple Text Sources Step One: Cross-Document Structure. In: Proceedings of the 1st SIGdial workshop on Discourse and dialogue. Association for Computational Linguistics, Morristown, NJ, USA, pp. 74–83. 13

Eberhart, R. C., Shi, Y. & Kennedy, J. (2001) Swarm Intelligence. The Morgan Kaufmann Series in Artificial Intelligence. Morgan Kaufmann. 34

Faruqui, M. & Padó, S. (2010) Training and Evaluating a German Named Entity Recognizer with Semantic Generalization. In: Proceedings of KONVENS 2010. Saarbrücken, Germany. 108

Fellbaum, C. (1998) WordNet An Electronic Lexical Database. The MIT Press, Cambridge, MA ; London. 18

Ferrández, Ó., Muñoz, R. & Palomar, M. (2009) Alicante University at TAC 2009: Experiments in RTE. In: Proceedings of TAC 2009. 19

Figueroa, A. & Neumann, G. (2008) Genetic Algorithms for Data-Driven Web Question Answering. Evolutionary Computation 16(1):89–125. 42

Finkel, J., Grenager, T. & Manning, C. (2005) Incorporating Non-Local Information into Information Extraction Systems by Gibbs Sampling. In: Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 363–370. 108

Galanis, D. & Malakasiotis, P. (2009) Aueb at TAC 2008. In: Proceedings of TAC 2008. 18

Gamallo Otero, P. & Gonzalez Lopez, I. (2011) A Grammatical Formalism based on Patterns of Part of Speech tags. International journal of corpus linguistics 16(1):45–71. 93, 126

Giampiccolo, D., Magnini, B., Dagan, I. & Dolan, B. (2007) The Third PASCAL Recognizing Textual Entailment Challenge. In: Proceedings of the ACLPASCAL Workshop on Textual Entailment. 15, 62

Giampiccolo, D., Dang, H. T., Magnini, B., Dagan, I. & Cabrio, E. (2008) The Fourth PASCAL Recognizing Textual Entailment Challenge. In: Proceedings of the ACLPASCAL Workshop on Textual Entailment. 15

Giménez, J. & Màrquez, L. (2007) Linguistic Features for Automatic Evaluation of Heterogenous MT Systems. In: Proceedings of the Second Workshop on Statistical Machine Translation. Association for Computational Linguistics, pp. 256–264. 120, 127

Giuliano, C. (2007) jLSI a Tool for Latent Semantic Indexing. Software available at http://tcc. itc. it/research/textec/toolsresources/jLSI. html . 56, 80

Glickman, O. (2006) Applied Textual Entailment. Ph.D. thesis, Department of Computer Science - Bar Ilan University. 12

Glickman, O., Dagan, I. & Koppel, M. (2006) A Lexical Alignment Model for Probabilistic Textual Entailment. Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Textual Entailment pp. 287–298. 30

Harabagiu, S. & Hickl, A. (2006) Methods for Using Textual Entailment in Open-Domain Question Answering. In: Proceedings of ACL'06. Associa-

tion for Computational Linguistics, Morristown, NJ, USA, pp. 905–912. 12, 65

Harmeling, S. (2009) Inferring Textual Entailment with a Probabilistically Sound Calculus. Natural Language Engineering 15(04):459–477. 29

Haussler, D. (1999) Convolution Kernels on Discrete Structures. Tech. rep. 54

Iftene, A. & Balahur-Dobrescu, A. (2007) Hypothesis Transformation and Semantic Variability Rules Used in Recognizing Textual Entailment. ACL 2007 p. 125. 30

Iftene, A. & Moruz, M. (2009) Uaic participation at RTE-5. Proceedings of TAC, Gaithersburg, Maryland . 30

Iordanskaja, L., Kittredge, R. & Polguere, A. (1991) Lexical Selection and Paraphrase in a Meaning-Text Generation Model. Natural language generation in artificial intelligence and computational linguistics 312. 25, 86

Jiang, J. J. & Conrath, D. W. (1997) Semantic Similarity based on Corpus Statistics and Lexical Taxonomy. In: Proc. of the 10th ROCLING. Tapei, Taiwan, pp. 132–139. 56

Joachims, T. (1999a) Making Large-scale Support Vector Machine Learning Practical, MIT Press, Cambridge, MA, USA, pp. 169–184. 26, 78, 87, 108

Joachims, T. (1999b) Making Large-Scale Support Vector Machine Learning Practical . 56

Joachims, T. (1999c) Svmlight: Support Vector Machine. SVM-Light Support Vector Machine http://svmlight. joachims. org/, University of Dortmund 19. 130

Klein, D. & Manning, C. D. (2003) Fast Exact Inference with a Factored Model for Natural Language Parsing. In: Advances in Neural Information Processing Systems 15. MIT Press, Cambridge, MA, pp. 3–10. 37

Koehn, P., Och, F. J. & Marcu, D. (2003) Statistical Phrase-based Translation. In: NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology. Association for Computational Linguistics, Morristown, NJ, USA, pp. 48–54. 85

Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zens, R., Dyer, C., Bojar, O., Constantin, A. & Herbst, E. (2007) Moses: Open Source Toolkit for Statistical Machine Translation. In: Proceedings of ACL07 Demo and Poster Sessions. 86, 95, 107, 108, 127

Kotb, Y. (2006) Toward Efficient Peer-to-Peer Information Retrieval Based on Textual Entailment. Web Intelligence and Intelligent Agent Technology, International Conference on 0:455–458. 13

Kouylekov, M. & Magnini, B. (2005) Recognizing Textual Entailment with Tree Edit Distance Algorithms. In: Proceedings of the First Challenge Workshop Recognising Textual Entailment. pp. 17–20. 29, 32, 33

Kouylekov, M. & Magnini, B. (2006) Tree Edit Distance for Recognizing Textual Entailment: Estimating the Cost of Insertion. In: Proc. of the PASCAL RTE-2 Challenge. pp. 68–73. 33

Kouylekov, M. & Negri, M. (2010) An Open-Source Package for Recognizing Textual Entailment. In: Proceedings of the ACL 2010 System Demonstrations. Association for Computational Linguistics, Stroudsburg, PA, USA, ACLDemos '10. 6, 22, 29, 32, 37, 63

Kouylekov, M., Mehdad, Y. & Negri, M. (2010a) Mining Wikipedia for Large-Scale Repositories of Context-Sensitive Entailment Rules. In: Proceedings of the Language Resources and Evaluation Conference (LREC 2010). 7, 19, 49, 62, 70, 80, 171

Kouylekov, M., Mehdad, Y., Negri, M. & Cabrio, E. (2010b) FBK Participation in RTE-6: Main and KBP Validation Task. In: Proceedings of the Sixth Recognizing Textual Entailment Challenge. 46

Kouylekov, M., Mehdad, Y. & Negri, M. (2011) Is it Worth Submitting this Run? Assess your RTE System with a Good Sparring Partner. TextInfer 2011 EMNLP Workshop on Textual Entailment p. 30. 6, 29, 60, 110, 171

Landauer, Foltz & Laham (1998) Introduction to Latent Semantic Analysis. In: Discourse Processes 25. 50

Li, F., Zheng, Z., Bu, F., Tang, Y., Zhu, X. & Huang, M. (2009a) Quanta at TAC 2009 KBP and RTE track. In: the Text Analysis Conference (TAC 2009) Workshop. Gaithersburg, Maryland, USA. 19

Li, F., Zheng, Z., Bu, F., Tang, Y., Zhu, X. & Huang, M. (2009b) Thu QUANTA at TAC 2009 KBP and RTE Track. In: Text Analysis Conference (TAC). 30

Lin, C. (2003) ROUGE: Recall-oriented Understudy for Gisting Evaluation. 119

Lin, D. (1998) Automatic Retrieval and Clustering of Similar Words. In: Proceedings of the 17th international conference on Computational linguistics-Volume 2. Association for Computational Linguistics, pp. 768–774. 23

Lin, D. & Pantel, P. (2001) DIRT-Discovery of Inference Rules from Text. Knowledge Discovery and Data Mining 323328. 20

Lloret, E., Ferrández, Ó., Muñoz, R. & Palomar, M. (2008) A Text Summarization Approach under the Influence of Textual Entailment. In: Proceedings of NLPCS 2008. 65

MacCartney, B. & Manning, C. (2007) Natural Logic for Textual Inference. In: Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. Association for Computational Linguistics, pp. 193–200. 28

Malakasiotis, P. (2009) Aueb at TAC 2009. In: Proc. of the Text Analysis Conference, Gaithersburg, MD. 30

McKeown, K., Barzilay, R., Evans, D., Hatzivassiloglou, V., Klavans, J., Nenkova, A., Sable, C., Schiffman, B. & Sigelman, S. (2002) Tracking and Summarizing News on a Daily Basis with Columbia's Newsblaster. In: Proceedings of the second international conference on Human Language Technology Research. Morgan Kaufmann Publishers Inc., pp. 280–285. 25, 86

Mehdad, Y. (2009) Automatic Cost Estimation for Tree Edit Distance Using Particle Swarm Optimization. In: Proceedings of the ACL-IJCNLP 2009 Conference. 6, 29, 63, 171

Mehdad, Y. & Magnini, B. (2009a) Optimizing Textual Entailment Recognition Using Particle Swarm Optimization. In: Proceedings of the ACL09 Workshop on Applied Textual Inference. 6, 29, 74, 171

Mehdad, Y. & Magnini, B. (2009b) A Word Overlap Baseline for the Recognizing Textual Entailment Task. 38

Mehdad, Y. & Magnini, B. (2009c) A word Overlap Baseline for the Recognizing Textual Entailment Task. 71

Mehdad, Y. & Negri, M. (2012) FBK: Cross-Lingual Textual Entailment without Translation. In: Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012). 9, 115, 172

Mehdad, Y., Moschitti, A. & Zanzotto, F. (2009a) SemKer: Syntactic/Semantic Kernels for Recognizing Textual Entailment. In: TAC 2009 Workshop. 22

Mehdad, Y., Negri, M., Cabrio, E., Kouylekov, M. & Magnini, B. (2009b) EDITS: An Open Source Framework for Recognizing Textual Entailment. In: Proceedings of TAC 2009. To appear. 19, 22, 74

Mehdad, Y., Moschitti, A. & Zanzotto, F. M. (2010a) Syntactic/Semantic Structures for Textual Entailment Recognition. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. Association for Computational Linguistics, Stroudsburg, PA, USA, HLT '10. 7, 8, 19, 22, 63, 171

Mehdad, Y., Negri, M. & Federico, M. (2010b) Towards Cross-Lingual Textual Entailment. In: Proceeding of short papers in NAACL 2010. 8, 106, 171

Mehdad, Y., Negri, M. & Federico, M. (2011) Using Bilingual Parallel Corpora for Cross-Lingual Textual Entailment. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1. Association for Computational Linguistics, pp. 1336–1345. 8, 20, 27, 62, 106, 110, 172

Mehdad, Y., Negri, M. & Federico, M. (2012a) Detecting Semantic Equivalence and Information Disparity in Cross-lingual Documents. In: Proceedings of the ACL'12. 9, 116, 172

Mehdad, Y., Negri, M. & Federico, M. (2012b) Match without a Referee: Evaluating MT Adequacy without Reference Translations. In: Proceedings of the Machine Translation Workshop (WMT2012). 9, 118, 139, 172

Melamed, I., Green, R. & Turian, J. (2003) Precision and Recall of Machine Translation. In: Proceedings of HLT-NAACL 2003–short papers-Volume 2. Association for Computational Linguistics, pp. 61–63. 119

Melgani, F. & Bazi, Y. (2008) Classification of Electrocardiogram Signals With Support Vector Machines and Particle Swarm Optimization. IEEE Transactions on Information Technology in Biomedicine 12(5):667–677. 35, 38

Mihalcea, R. & Strapparava, C. (2009) The Lie Detector: Explorations in the Automatic Recognition of Deceptive Language. In: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers. Association for Computational Linguistics, pp. 309–312. 154

Mihalcea, R., Corley, C. & Strapparava, C. (2006) Corpus-based and Knowledge-based Measures of Text Semantic Similarity. In: Proceedings of AAAI06. 51

Mirkin, S., Bar-Haim, R., Berant, J., Dagan, I., Shnarch, E., Stern, A. & Szpektor, I. (2009a) Addressing Discourse and Document Structure in the RTE Search Task. Proc. of TAC . 20

Mirkin, S., Specia, L., Cancedda, N., Dagan, I., Dymetman, M. & Szpektor, I. (2009b) Source-Language Entailment Modeling for Translating Unknown Terms. In: Proceedings of ACL '09. 3, 14

Morik, K., Brockhausen, P. & Joachims, T. (1999) Combining Statistical Learning with a Knowledge-based Approach-a Case Study in Intensive

Care Monitoring. In: Machine Learning International Workshop. MORGAN KAUFMANN Publishers, INC., pp. 268–277. 58

Moschitti, A. (2006) Making Tree Kernels Practical for Natural Language Learning. In: Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics. 56

Mrozinski, J., Whittaker, E. & Furui, S. (2008) Collecting a Why-Question Corpus for Development and Evaluation of an Automatic QA-System. Proceedings of ACL-08: HLT pp. 443–451. 154

Negri, M. & Mehdad, Y. (2010) Creating a Bi-Lingual Entailment Corpus through Translations with Mechanical Turk: $100 for a 10-day Rush. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazonś Mechanical Turk. Association for Computational Linguistics, pp. 212–216. 9, 153, 173

Negri, M., Kouylekov, M., Magnini, B., Mehdad, Y. & Cabrio, E. (2009) Towards Extensible Textual Entailment Engines: the EDITS Package. In: AI* IA 2009: Emergent Perspectives in Artificial Intelligence. Springer, pp. 314–323. 37

Negri, M., Bentivogli, L., Mehdad, Y., Giampiccolo, D. & Marchetti, A. (2011) Divide and Conquer: Crowdsourcing the Creation of Cross-Lingual Textual Entailment Corpora. Proc. of EMNLP 2011 . 9, 105, 114, 173

Negri, M., Marchetti, A., Mehdad, Y., Bentivogli, L. & Giampiccolo, D. (2012) Semeval-2012 Task 8: Cross-lingual Textual Entailment for Content Synchronization. In: Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012). 101, 173

Neuhaus, M. & Bunke, H. (2004) A Probabilistic Approach to Learning Costs for Graph Edit Distance. In: Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on. IEEE, vol. 3, pp. 389–393. 33

Nielsen, R., Becker, L. & Ward, W. (2009) 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. 31

Nießen, S., Och, F., Leusch, G. & Ney, H. (2000) An Evaluation Tool for Machine Translation: Fast Evaluation for MT Research. In: Proceedings of the 2nd International Conference on Language Resources and Evaluation. pp. 39–45. 119

Och, F. & Ney, H. (2000) Improved Statistical Alignment Models. In: Proceedings of the 38th Annual Meeting on Association for Computational Linguistics. Association for Computational Linguistics, pp. 440–447. 85, 95, 107, 108, 127

Padó, S. (2006) User's Guide to `sigf`: Significance Testing by Approximate Randomisation. 111

Padó, S., de Marneffe, M., MacCartney, B., Rafferty, A., Yeh, E. & Manning, C. (2008) Deciding Entailment and Contradiction with Stochastic and Edit Distance-Based Alignment. In: Text Analysis Conference (TAC 2008) Workshop-RTE-4 Track. National Institute of Standards and Technology. pp. 17–19. 30

Padó, S., Galley, M., Jurafsky, D. & Manning, C. (2008) Evaluation of MT Output with Entailment Technology. In: NIST Metrics MATR'08 system description. 13

Padó, S., Galley, M., Jurafsky, D. & Manning, C. D. (2009) Textual Entailment Features for Machine Translation Evaluation. In: Proceedings of StatMT '09. 120, 123

Papineni, K., Roukos, S., Ward, T. & Zhu, W. (2002) BLEU: a Method for Automatic Evaluation of Machine Translation. In: Proceedings of the 40th annual meeting on association for computational linguistics. Association for Computational Linguistics, pp. 311–318. 119

Pedersen, T., Patwardhan, S. & Michelizzi, J. (2004) WordNet::Similarity - Measuring the Relatedness of Concepts. In: Proc. of 5th NAACL. 49, 51, 56, 61

Porter, M. (2001) Snowball: A Language for Stemming Algorithms. 26, 77

Quirk, C. B. (2004) Training a Sentence-Level Machine Translation Confidence Measure. In: Proceedings of LREC. 123

Rafferty, A. & Manning, C. (2008) Parsing Three German Treebanks: Lexicalized and Unlexicalized Baselines. In: Proceedings of the Workshop on Parsing German. Association for Computational Linguistics, pp. 40–46. 107

Rodrigo, A., Penas, A. & Verdejo, F. (2008) Towards an Entity-based Recognition of Textual Entailment. In: Proceedings of the Fourth PASCAL Challenges Workshop on Recognizing Textual Entailment. Gaithersburg, Maryland, USA. 31

Rodríguez, L., García-Varea, I. & Gámez, J. (2008) On The Application of Different Evolutionary Algorithms to the Alignment Problem in Statistical Machine Translation. Neurocomputing 71(4):755–765. 42

Romano, L., Kouylekov, M., Szpektor, I., Dagan, I. & Lavelli, A. (2006)

Investigating a Generic Paraphrase-based Approach for Relation Extraction. In: Proceedings of EACL 2006. 65

Roth, D., Sammons, M. & Vydiswaran, V. (2009) A Framework for Entailed Relation Recognition. In: Proceedings of the ACL-IJCNLP 2009 Conference Short Papers. 3, 13

Roy Bar-Haim, I., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B. & Szpektor, I. (2006) The Second PASCAL Recognising Textual Entailment Challenge. In: Proceedings of the ACLPASCAL Workshop on Textual Entailment and. 15, 62

Sammons, M., Vydiswaran, V., Vieira, T., Johri, N., Chang, M., Goldwasser, D., Srikumar, V., Kundu, G., Tu, Y., Small, K. et al. (2009) Relation Alignment for Textual Entailment Recognition. Proceedings of Recognizing Textual Entailment 2009 . 30

Sammons, M., Vydiswaran, V. & Roth, D. (2010) Ask Not What Textual Entailment Can Do For You... In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 1199–1208. 17

Schmid, H. (1995) TreeTagger a Language Independent Part-Of-Speech Tagger. Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart p. 43. 26, 77

Schuler, K. K. (2005) Verbnet: a Broad-Coverage, Comprehensive Verb Lexicon. Ph.D. thesis, University of Pennsylvania, Philadelphia, PA, USA. 19

Shnarch, E. (2008) Lexical Entailment and its Extraction from Wikipedia. Master's thesis, Bar-Ilan University, Israel. 19

Siblini, R. & Kosseim, L. (2008) Using Ontology Alignment for the TAC RTE Challenge. In: Text Analysis Conference (TAC 2008) Workshop-RTE-4 Track. National Institute of Standards and Technology. pp. 17–19. 30

Snover, M., Dorr, B., Schwartz, R., Micciulla, L. & Makhoul, J. (2006) A Study of Translation Edit Rate with Targeted Human Annotation. In: Proceedings of Association for Machine Translation in the Americas. pp. 223–231. 119

Snover, M., Madnani, N., Dorr, B. & Schwartz, R. (2009) Fluency, Adequacy, or HTER? Exploring Different Human Judgements with a Tunable MT Metric. In: Proceedings of the Fourth Workshop on Statistical Machine Translation. Association for Computational Linguistics, vol. 30, pp. 259–268. 25, 86

Snow, R., O'Connor, B., Jurafsky, D. & Ng, A. (2008) Cheap and Fast—but is it Good?: Evaluating Non-Expert Annotations for Natural Language Tasks. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, pp. 254–263. 144, 147

Specia, L. (2011) Exploiting Objective Annotations for Measuring Translation Post-Editing Effort. In: 15th Conference of the European Association for Machine Translation. pp. 73–80. 123

Specia, L. & Farzindar, A. (2010) Estimating Machine Translation Post-Editing Effort with HTER. In: AMTA-2010 Workshop Bringing MT to the User: MT Research and the Translation Industry. pp. 33–41. 123

Specia, L., Cancedda, N., Dymetman, M., Turchi, M. & Cristianini, N. (2009) Estimating the Sentence-Level Quality of Machine Translation

Systems. In: The 13th Annual Conference of the European Association for Machine Translation (EAMT-2009). pp. 28–35. 123, 124, 131, 132, 134, 137

Specia, L., Cancedda, N. & Dymetman, M. (2010) A Dataset for Assessing Machine Translation Evaluation Metrics. In: Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC10), Valletta, Malta. European Language Resources Association (ELRA). 129

Specia, L., Hajlaoui, N., Hallett, C. & Aziz, W. (2011) Predicting Machine Translation Adequacy. In: Proceedings of the Thirteenth Machine Translation Summit (MTSummit-2011), China. 123, 133, 135, 138

Szpektor, I. & Dagan, I. (2008) Learning Entailment Rules for Unary Templates. In: Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1. Association for Computational Linguistics, pp. 849–856. 20

Tatu, M. & Moldovan, D. (2005) A Semantic Approach to Recognizing Textual Entailment. In: Proceedings of HLT/EMNLP 2005. pp. 371–378. 19

Tatu, M. & Moldovan, D. (2007) COGEX at RTE-3. In: Proceedings of ACL-07. 19, 27

Tatu, M., Iles, B., Slavick, J., Novischi, A. & Moldovan, D. (2006) Cogex at the Second Recognizing Textual Entailment Challenge. In: Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment. pp. 104–109. 27

Tillmann, C., Vogel, S., Ney, H., Zubiaga, A. & Sawaf, H. (1997) Accel-

erated DP Based Search for Statistical Translation. In: Fifth European Conference on Speech Communication and Technology. 119

Van Rijsbergen, C. J. (1979) Information Retrieval, 2nd edition. Dept. of Computer Science, University of Glasgow. 13

Wang, R. (2011) Intrinsic and Extrinsic Approaches to Recognizing Textual Entailment. Saarbrücken dissertations in computational linguistics and language technology. German Research Center for Artifical Intelligence. 30

Wang, R. & Callison-Burch, C. (2010) Cheap Facts and Counter-Facts. In: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk. Association for Computational Linguistics, pp. 163–167. 153, 154

Wang, R. & Neumann, G. (2008a) An Accuracy-Oriented Divide-and-Conquer Strategy for Recognizing Textual Entailment. In: Proceedings of Text Analysis Conference. Gaithersburg, Maryland USA. 31

Wang, R. & Neumann, G. (2008b) Information Synthesis for Answer Validation. In: CLEF 2008 Working Notes. Springer Verlag. 13

Wang, R. & Neumann, G. (2008c) Relation Validation via Textual Entailment. In: 1st International and KI-08 Workshop on Ontology-based Information Extraction Systems. pp. 26–37. 13

Wu, Z. & Palmer, M. (1994) Verb Semantics And Lexical Selection. In: Proceedings of the 32nd Annual Meeting of the ACL. 50

Xiong, D., Zhang, M. & Li, H. (2010) Error Detection for Statistical Machine Translation using Linguistic Features. In: Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, pp. 604–611. 123

Yatbaz, M. (2008) RTE-4: Normalized Dependency Tree Alignment Using Unsupervised N-gram Word Similarity Score. In: Proceedings of the Fourth PASCAL Challenges Workshop on Recognizing Textual Entailment. Gaithersburg, Maryland, USA. 30

Zanzotto, F. M. & Moschitti, A. (2006a) Automatic Learning of Textual Entailments with Cross-Pair Similarities. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics. 31

Zanzotto, F. M. & Moschitti, A. (2006b) Automatic Learning of Textual Entailments with Cross-Pair Similarities. In: Proceeding of ACL '06. 48, 54

Zanzotto, F. M., Pennacchiotti, M. & Moschitti, A. (2009) A Machine Learning Approach to Textual Entailment Recognition. Natural Language Engineering . 48, 54, 57

Zhang, C. & Chai, J. Y. (2010) Towards Conversation Entailment: An Empirical Investigation. In: EMNLP. 3, 14

Zhao, S., Wang, H., Liu, T. & Li, S. (2009) Extracting Paraphrase Patterns from Bilingual Parallel Corpora. Natural Language Engineering 15(04):503–526. 70

Appendix A

List of Published Papers

2012

1. Yashar Mehdad, Matteo Negri and Marcello Federico. *Detecting Semantic Equivalence and Information Disparity in Cross-lingual Documents.* In **ACL** 2012 - The 50th Annual Meeting of the Association for Computational Linguistics".

2. Yashar Mehdad, Matteo Negri and Marcello Federico. *Match without a Referee: Evaluating MT Adequacy without Reference Translations.* In 7th Workshop on Statistical Machine Translation **WMT** 2012. Montreal Canada.

3. Matteo Negri, Alessandro Marchetti, Yashar Mehdad, Luisa Bentivogli and Danilo Giampiccolo. *Semeval-2012 Task 8: Cross-lingual Textual Entailment for Content Synchronization.* Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012), in conjunction with the First Joint Conference on Lexical and Computational Semantcis (*SEM 2012). Montreal Canada. 2012.

4. Yashar Mehdad, Matteo Negri and José Guilherme Camargo de Souz. *FBK: Cross-Lingual Textual Entailment without Translation.* Proceedings of the 6th International Workshop on Semantic Evalua-

tion (SemEval 2012), in conjunction with the First Joint Conference on Lexical and Computational Semantcis (*SEM 2012). Montreal Canada.

5. José Guilherme Camargo de Souz, Matteo Negri and Yashar Mehdad. *FBK: Combining Machine Translation Evaluation and Word Similarity metrics for Semantic Textual Similarity.* Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012), in conjunction with the First Joint Conference on Lexical and Computational Semantcis (*SEM 2012). Montreal Canada.

2011

1. Y. Mehdad, M. Negri, and M. Federico. *Using bilingual parallel corpora for cross-lingual textual entailment.* In Proceedings of the **ACL** 2011 Conference. The Association for Computer Linguistics, 2011

2. M. Negri, L. Bentivogli, Y. Mehdad, D. Giampiccolo, and A. Marchetti. *Divide and conquer: Crowdsourcing the creation of cross-lingual textual entailment corpora.* In Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (**EMNLP**), Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics.

3. M. Kouylekov, Y. Mehdad, and M. Negri. *Is it worth submitting this run? assess your RTE system with a good sparring partner.* In Proceedings of the **TextInfer** 2011 Workshop on Textual Entailment, Edinburgh, Scottland, UK, July 2011. Association for Computational Linguistics.

4. C. Monz, V. Nastase, M. Negri, A. Fahrni, Y. Mehdad, and M. Strube. *CoSyne: A framework for multilingual content synchronization of Wikis.* In Proceedings of the 7th International Symposium

on Wikis and Open Collaboration (WikiSym), Mountain View, Calif., 3-5 October 2011.

2010

1. Y. Mehdad, A. Moschitti, and F.M. Zanzotto. *Syntactic/semantic structures for textual entailment recognition.* In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (**NAACL**), 2010.

2. M. Negri and Y. Mehdad. *Creating a bi-lingual entailment corpus through translations with mechanical turk: $100 for a 10-day rush.* In Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, pages 212-216, 2010.

3. Y. Mehdad, M. Negri, and M. Federico. *Towards cross-lingual textual entailment.* In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (**NAACL**), pages 321-324. Association for Computational Linguistics, 2010.

4. M. Kouylekov, Y. Mehdad, and M. Negri. *Mining wikipedia for large-scale repositories of context-sensitive entailment rules.* In Proceedings of the Language Resources and Evaluation Conference (LREC 2010), 2010.

5. M. Kouylekov, Y. Mehdad, M. Negri, and E. Cabrio. *Fbk participation in rte6: Main and kbp validation task.* In Proceedings of the Sixth Recognizing Textual Entailment Challenge, 2010.

2009

1. Y. Mehdad. *Automatic Cost Estimation for Tree Edit Distance Using Particle Swarm Optimization.* In Proceedings of the **ACL**-IJCNLP 2009 Conference Short Papers, pages 289-292. Association for Computational Linguistics, 2009.

2. Y. Mehdad and B. Magnini. *Optimizing textual entailment recognition using particle swarm optimization.* In Proceedings of the 2009 Workshop on Applied Textual Inference, **TextInfer** '09. Association for Computational Linguistics, 2009.

3. Y. Mehdad, A. Moschitti, and F.M. Zanzotto. Semker: *Syntactic/semantic kernels for recognizing textual entailment.* In Proc. of the Text Analysis Conference, Gaithersburg, MD, 2009.

4. M. Negri, M. Kouylekov, B. Magnini, Y. Mehdad, and E. Cabrio. *Towards extensible textual entailment engines: the EDITS package.* In AI* IA 2009: Emergent Perspectives in Artificial Intelligence, pages 314-323. Springer, 2009.

5. Y. Mehdad, V. Scurtu, and E. Stepanov. *Italian named entity recognizer participation in NER task@ Evalita 09.* In Proceedings of EVALITA'09, 2009.

6. E. Cabrio, Y. Mehdad, M. Negri, M. Kouylekov, and B. Magnini. *Recognizing textual entailment for Italian EDITS@ Evalita'09.* In Proceedings of the EVALITA, 2009.

7. Y. Mehdad and B. Magnini. *A word overlap baseline for the recognizing textual entailment task*, 2009.

8. Y. Mehdad, M. Negri, E. Cabrio, M. Kouylekov, and B. Magnini. *Using lexical resources in a distance-based approach to rte.* In Proceedings of the TAC 2009 Workshop on Textual Entailment, 2009.