PhD Dissertation

International Doctorate School in Information and Communication Technologies

DISI - University of Trento

Cross-Domain and Cross-Language Porting of Shallow Parsing

Evgeny A. Stepanov

Advisor:

Prof. Dr. Ing. Giuseppe Riccardi

Università degli Studi di Trento

April 2014

Abstract

English was the main focus of attention of the Natural Language Processing (NLP) community for years. As a result, there are significantly more annotated linguistic resources in English than in any other language. Consequently, data-driven tools for automatic text or speech processing are developed mainly for English. Developing similar corpora and tools for other languages is an important issue. However, this requires significant amount of effort. Recently, Statistical Machine Translation (SMT) techniques and parallel corpora were used to transfer annotations from a linguistic resource rich languages to a resource-poor languages for a variety of Natural Language Processing (NLP) tasks, including Part-of-Speech tagging, Noun Phrase chunking, dependency parsing, textual entailment, etc.

This cross-language NLP paradigm relies on the solution of the following sub-problems:

1. *Data-driven NLP techniques are very sensitive to the differences in training and testing conditions. Different domains, such as financial news-wire and biomedical publications, have different distributions of NLP task-specific properties; thus, the* ***domain adaptation*** *of the source language tools – either the development of models with good cross-domain performance or tuned to the target domain – is critical.*
2. *Another difference in training and testing conditions arises with cross-genre applications such as written text (monologues) and spontaneous dialog data. Properties of written text such as punctuation and the notion of sentence are not present in spoken conversation transcriptions. Thus,* ***style-adaptation*** *techniques to cover a wider range of genres is critical as well.*
3. *The basis of cross-language porting is parallel corpora. Unfortunately, parallel corpora are scarce. Thus, generation or retrieval of parallel corpora between the languages of interest is important. Addition-*

ally, these parallel corpora most often are not in the domains of interest; consequently, the cross-language porting should be augmented with SMT domain adaptation techniques.

4. The language distance play an important role within the paradigm, since for close family language pairs (e.g. Romance languages Italian and Spanish) the range of linguistic phenomena to consider is significantly less compared to the distant family language pairs (e.g. Italian and Turkish). The developed cross-language techniques should be applicable to both conditions.

In this thesis we address these sub-problems on complex Natural Language Processing tasks of Discourse Parsing and Spoken Language Understanding. Both tasks are cast as ***token-level shallow parsing***.

Penn Discourse Treebank (PDTB) style discourse parsing is applied cross-domain and we contribute feature-level domain adaptation techniques for the task. Additionally, we explore PDTB-style discourse parsing on dialog data in Italian are report on challenges. The problems of parallel corpora creation, language style adaptation, SMT domain-adaptation and language distance are addressed on the task of cross-language porting of Spoken Language Understanding.

This thesis contributes to the task with the language-style and domain adaptation techniques for machine translation of spoken conversations using off-the-shelf systems like Google Translate, SMT systems trained on both out-of-domain and in-domain parallel data. We demonstrate that the techniques are beneficial for both close and distant language pairs. We propose the methodologies for the creation of parallel spoken conversation corpora via professional translation services that considers speech phenomena such as disfluencies. Additionally, we explore the semantic annotation transfer using automatic SMT methods and crowdsourcing. For the later, we propose the computational methodology to obtain acceptable quality corpus without the target language references and the low worker agreement.

Keywords

Shallow Parsing, Cross-Language Porting, Statistical Machine Translation, Language Style Adaptation, Domain Adaptation

Contents

List of Tables

List of Figures

Chapter 1

Introduction

English was the main focus of attention of the Natural Language Processing (NLP) community for years. As a result, there are significantly more annotated linguistic resources in English than in any other language. Consequently, data-driven tools for automatic text or speech processing are developed mainly for English. Developing similar corpora and tools for other languages is an important issue. However, this requires significant amount of effort. Recently, Statistical Machine Translation (SMT) techniques and parallel corpora were used to transfer annotations from a linguistic resource rich languages to a resource-poor languages for a variety of Natural Language Processing (NLP) tasks. In this thesis we address the problems related to cross-language porting of NLP applications.

1.1 The Problem

Starting from yearly 2000's, parallel corpora has been used for cross-lingual Natural Language Processing (NLP). One of the uses is *annotation projection* [78]. The main goal of the approach is leveraging the effort spent on English (or some other resource-rich language) to create comparable size annotated monolingual corpora in some other resource-poor language; ultimately training NLP tools in these languages. The annotation transfer was done via statistical word alignments and the annotation noise was compensated by robust statistical learning algorithms.

The alignment projection approach was successfully applied to create monolingual annotated data for a variety of linguistic phenomena. In

[78], the authors transferred annotations from English to close and distant languages and created resources for Part-of-Speech tagging (also [75]), Noun Phrase chunking, Named-Entity tagging and morphological analysis. Other applications include dependency parsing [28], temporal information [62], word sense information [4], information extraction [59], FrameNet [48], and others.

This cross-language NLP paradigm relies on the solution of the following sub-problems:

- Data-driven NLP techniques are very sensitive to the differences in training and testing conditions. Different domains, such as financial news-wire and biomedical publications, have different distributions of NLP task-specific properties; thus, the **domain adaptation** of the source language tools – either the development of models with good cross-domain performance or tuned to the target domain – is critical.

- Another difference in training and testing conditions arises with cross-genre applications such as written text (monologues) and spontaneous dialog data. Properties of written text such as punctuation and the notion of sentence are not present in spoken conversation transcriptions. Thus, **style-adaptation** techniques to cover a wider range of genres is critical as well.

- The basis of cross-language porting is parallel corpora. Unfortunately, parallel corpora are scarce. Thus, generation or retrieval of parallel corpora between the languages of interest is important. Additionally, these parallel corpora most often are not in the domains of interest; consequently, the cross-language porting should be augmented with SMT domain adaptation techniques.

- The language distance plays an important role within the paradigm, since for close family language pairs (e.g. Romance languages such as Italian and Spanish) the range of linguistic phenomena to consider is significantly less compared to the distant family language pairs (e.g. Italian and Turkish). The developed cross-language techniques should be applicable to both conditions.

1.2 The Background

In this thesis we address these sub-problems on complex Natural Language Processing tasks of Discourse Parsing and Spoken Language Understanding. Both tasks are cast as **token-level shallow parsing** with Conditional Random Fields [38].

Discourse analysis is one of the most challenging tasks in Natural Language Processing, that has applications in many language technology areas such as opinion mining, summarization, information extraction, etc. [73, 69]. With the availability of annotated corpora, such as Penn Discourse Treebank (PDTB) [52], statistical discourse parsers were developed [42, 22, 76]. The output of a PDTB-style discourse parser usually is a discourse relation – a triplet of a discourse connective and its two arguments – annotated with a discourse relation sense. This information is indubitably useful for many NLP tasks. Unfortunately, discourse relation annotated resources are scarce: available only for a handful of languages and are mainly written monologues. This significantly constrains their applicability.

Token-level sequence labeling with CRFs is a popular approach to Spoken Language Understanding (SLU), which is also referred to as *shallow* semantic parsing. The tasks are similar in that the CRFs utilize sets of features. However, the tasks differ in the label set: it is much larger for SLU. Additionally, there is a difference in the span size, which is shorter for SLU.

In the context of cross-language porting of speech applications, the problem takes an additional aspect. Since in speech applications it is the machine who will process the output of SLU, with respect to the direction and the object of translation the approaches to cross-language porting via Statistical Machine Translation (SMT) can be grouped into two categories.

The *Test-on-Target* approach (also referred to as Train-on-Target), relies on cross-lingual annotation projection that was described in the previous section. The direction of translation is from the source language to the target language; and the object of translation is the data used to train the source system. Ultimately, a new Natural Language Processing system is trained in the target language. The approach heavily relies on the accurate transfer of annotation.

In the *Test-on-Source* approach the direction of translation is from a language the system is being ported to (target language) to the language of the existing (source language) system. The object of translation is a document in the target language. The success of the approach depends on the quality of machine translation.

In the literature, the Test-on-Source approach to Spoken Language Understanding system porting is credited as having better performance (e.g. [29, 30, 31, 40]). Moreover, the procedure is simpler to implement, since it does not require porting of annotation. However, the approach is applicable only if the end goal is language agnostic: such as Spoken Language Understanding, where the output is passed to Dialog Manager (i.e. machine), or Textual Entailment, where the end goal is to get a Yes/No response.

1.3 Contributions

The contributions of this thesis can be roughly partitioned as contributions to PDTB-style discourse parsing and cross-language Spoken Language Understanding system porting.

One of the contributions of this thesis is the re-structuring and re-evaluation of the discourse relation parser of [22]. We have compared the two approaches to discourse parsing – treating inter- and intra-sentential relations identically and separately – and re-structured the discourse parser cast as token-level sequence labeling with CRFs to process these two sets of relations differently. The re-structuring is motivated by the fact that heuristic-based argument span extraction of inter-sentential discourse relation yields better performance than the CRF models treating both relation types identically.

Additionally, we have changed the argument span extraction evaluation method of [22] from CONLL-based to string-based, since the CONLL-based evaluation yields does not take into account the existence of non-contiguous argument spans. This allows for more accurate comparison to other approaches to PDTB-style discourse parsing.

For the first time the PDTB-style discourse parser has been evaluated cross-domain on argument span extraction subtask. The PDTB-trained argument span extraction models have been evaluated on BioDRB and it was shown that the task cast as token-level sequence labeling has a good cross-domain generalization.

On top of the dependence of argument span extraction on discourse connective detection, the original features sets for the task included relation senses, so the task was depending on two discourse parsing subtasks. Unfortunately, discourse connective detection and relation sense classification both have poor cross-domain generalization. We applied feature-level domain adaptation and lifted the dependence from relation sense classification, while maintaining comparable performance.

We have contributed to the field of cross-language system porting with methodologies for utilizing out-of-domain data and generic online translation systems and adapting the system pipelines to the conversational style and the system domain. We have evaluated the language-style and domain adaptation techniques within Statistical Machine Translation (SMT) pipeline and showed significant improvements.

The usual SMT system optimization to a specific task, such as Spoken Language Understanding, required Minimum Error Training using the evaluation metric of that task (Concept Error Rate). However, the procedure is very time and resource consuming. We have contributed to the optimization with the re-ranking of n-best list of translation hypotheses with the joined Recurrent Neural Network Language Model trained on reference word-concept pairs.

An alternative to the cross-language porting of language resource annotations via SMT is to transfer them via crowdsourcing. We have proposed the methods for cross-language porting of semantic annotation via priming the workers with the source language concepts. We have demonstrated that the combination of computational techniques and the variability of user annotations can yield acceptable quality semantically annotated language resource even with low inter-annotator agreement.

Additionally, for the first time we have applied PDTB-style discourse parsing to spontaneous dialog data and on the language other than English. We have discussed the related challenges and outlined the solutions.

1.4 The Structure of the Thesis

Due to the fact that the research presented in this thesis can be divided into fairly independent tasks, each of the chapters contains a more detailed descriptions of the problems as well as the related work sections. The chapters are organized as follows.

In Chapter 2 we present the Italian LUNA Corpus and the process of the development of Multilingual LUNA Corpus for spoken language system porting.

In Chapter 3 we describe discourse parsing cast as token-level sequence labeling using CRFs. We define the subtasks of discourse relation parsing and evaluate their complexities. The different algorithms are evaluated and the best system architecture is selected.

In Chapter 4 we present the results from the literature on cross-domain evaluation of the discourse parsing subtasks of discourse connective detection and relation sense classification. We provide cross-domain evaluation of the argument position classification and argument span extraction subtasks and propose the domain-adaptation technique targeted to reduce the dependence of argument span extraction on the other subtasks of discourse parsing.

In Chapter 5 we present the cross-language porting via Statistical Machine Translation results on the LUNA Corpus. We evaluate the effects of the in-domain, out-of-domain, and general-domain SMT systems on Spoken Language Understanding porting on Test-on-Source and Test-on-Target scenarios. The differences between the approaches, which are reported in the literature, are confirmed for both close and distance language families. The Test-on-Source approach is selected for adaptation experiments.

In Chapter 6 we present the alternative approach to cross-language porting of language resources: crowdsourced cross-language transfer of semantic annotation in domain-specific conversation corpus. We present the crowdsourced semantic annotation transfer task and demonstrate that the combination of computational techniques with the variability in human annotation can yield acceptable quality annotated resource.

In Chapter 7 we present experiments on language-style and SMT domain adaptation for the Test-on-Source scenario for cross-language SLU porting. We demonstrate that proposed techniques significantly improve the translation quality and cross-language SLU performance. Additionally, we present the task-specific translation optimization and demonstrate that the approach improves the SLU performance further.

In Chapter 8 we present the analysis of discourse annotation of spoken dialog in LUNA Corpus. We discuss the challenges related to discourse parsing of dialogs. Additionally, the developed techniques are applied to the task of Discourse Connective Detection.

In Chapter 9 we summarize the thesis and provide future directions of this line of research.

Chapter 2

Corpora

The thesis addresses the issues of domain adaptation on a Discourse Parsing (DP) task and the issues of cross-language porting for low-resource languages on a Spoken Language Understanding (SLU) task. This chapter describes the corpora that are used for each of the tasks: Penn Discourse Treebank (PDTB) and Biomedical Discourse Relation Bank (BioDRB) for domain adaptation; Italian LUNA Corpus and Multilingual LUNA Corpus, its derivative, for the cross-language porting task. For the Multilingual LUNA Corpus we additionally describe the process of its creation from Italian LUNA Corpus.

2.1 Penn Discourse Treebank

The Penn Discourse Treebank (PDTB) [52] is the largest discourse annotated corpus. The discourse relation annotation is done on top of WSJ corpus (approximately 1M words); and it is aligned with Penn Treebank (PTB) syntactic tree annotation.

Besides its size, the other reasons that make PDTB appealing are that the discourse relations are *lexically-grounded* and *theory-neutral*. The lexical grounding is reflected in the fact that annotation is based on the presence of a discourse connective, or a possibility of its insertion. The theory neutrality, on the other hand, is reflected in the 'flat' nature of its annotation, which makes no assumptions on the high-level discourse structures (e.g. tree-like) [52]. Because of this theory-neutral approach, the PDTB-style discourse parsing is **shallow**.

Discourse relations in PDTB are binary: a discourse connective, considered as the predicate, takes exactly two arguments – *Arg1* and *Arg2* – where *Arg2* is the argument syntactically attached to the connective. With respect to *Arg2*, *Arg1* can appear in the same sentence (SS case), one of the preceding (PS case) or following (FS case) sentences. Both arguments can span over a part, one or more sentences. However, the annotator followed the 'minimality principle', and only portions of text minimally necessary for the interpretation of the discourse relation are included into argument span. Any other span that is relevant but not minimally necessary is annotated as *supplementary information.* Following argument naming conventions, text span supplementary to *Arg1* is labeled as *Sup1*, and to *Arg2* and *Sup2*.

Besides the discourse connective, its arguments, and their supplementary information, PDTB contains annotation of *attributions* of discourse relations. The relations are attributed to either writer (Wr), some other entity mentioned in text (Ot) or "arbitrary individual(s) indicated via non-specific reference" (Arb) [52]. Detection of attribution is an important task for discourse parsing, since it is a main factor that makes syntactic and discourse argument spans different [16]. However, it is out of the scope of this thesis.

2.1.1 PDTB Discourse Relation Types

A discourse connective is a member of a well defined list of 100 connectives and a relation expressed via such connective is an *Explicit* discourse relation. In the Example 1, the discourse connective 'But' (underlined) expresses *Comparison* relation between *Arg1* (in italics) and and *Arg2* (in bold).

However, a discourse relation can hold also without the presence of a connective. In PDTB adjacent sentences within a paragraph were additionally annotated for such *Implicit* discourse relations (see Example 2). In the implicit discourse relations, a connective can be inserted, but is left *implicit*, such as the connective 'for example'.

(1) *Country funds offer an easy way to get a taste of foreign stocks without the hard research of seeking out individual companies.*

Type	**Count**	**%**
Explicit	18,459	45.5%
Implicit	16,224	40.0%
AltLex	624	1.5%
EntRel	5,210	12.8%
NoRel	254	0.6%
Total	40,600	100.0%

Table 2.1: The distribution of annotated relation types in PDTB (from [52]).

But **it doesn't take much to get burned.**
(Explicit – Comparison:Contrast)

(2) *But it doesn't take much to get burned.*
Political and currency gyrations can whipsaw the funds.
(Implicit 'for example' – Expansion:Restatement)

In case a connective cannot be inserted while there is a discourse relation between sentences, the pair is annotated as having *Alternative Lexicalization* (AltLex) discourse relation. In the Example 3, 'Another concern' is an alternatively lexicalized connective expression.

(3) *Political and currency gyrations can whipsaw the funds.*
Another concern: **The funds' share prices tend to swing more than the broader market.**
(AltLex – Expansion:Conjunction)

In PDTB, in case an adjacent pair of sentences has neither explicit, nor implicit nor AltLex discourse relation, it is additionally inspected for whether the two sentences involve the same entity. Such sentences are annotated as having *Entity-based Coherence Relation* (EntRel). If the pair does not involve the same entity, it is annotated as *No Relation* (NoRel).

Table 2.1 gives the distribution of the annotated relations in PDTB. In this thesis we focus on *explicit* discourse relations only, which comprise 45.5% of all relations and 52.3% of all discourse relations in PDTB.

2.1.2 PDTB Discourse Relation Senses

In PDTB discourse relations are annotated using 3-level hierarchy of senses (see Table 2.2). The top level (level 1) senses are the most general: *Comparison*, *Contingency*, *Expansion*, and *Temporal* [52].

Class	Type	Sub-Type
Comparison	*Contrast*	juxtaposition opposition
	Pragmatic Contrast	
	Concession	expectation contra-expectation
	Pragmatic Concession	
Contingency	*Cause*	reason result
	Pragmatic Cause	justification
	Condition	hypothetical general unreal present unreal past factual present factual past
	Pragmatic Condition	relevance implicit assertion
Expansion	*Conjunction*	
	Instantiation	
	Restatement	specification equivalence generalization
	Alternative	conjunctive disjunctive chosen alternative
	Exception	
	List	
Temporal	*Synchronous*	
	Asynchronous	precedence succession

Table 2.2: PDTB discourse relation sense hierarchy (from [52]).

- *Comparison* relation expresses the differences between *Arg1* and *Arg2*;
- *Contingency* relation expresses the causality between *Arg1* and *Arg2*;
- *Expansion* relation expresses the continuation of elaboration from *Arg1* to *Arg2*, or vica versa;
- *Temporal* relation expresses the time-wise connection between *Arg1* and *Arg2*;

The second level of the connective sense hierarchy (types) contains 16 senses, which provided finer semantic distinctions, e.g. *Temporal* relations are further distinguished as *Synchronous and Asynchronous.* The third level of the hierarchy (sub-types) refines the second-level senses even further: e.g. *Temporal.Asynchronous* relations are further categorized as *precedence* and *succession.* For the experiments described in this thesis, however, we do not consider relation senses finer than the four top-level ones.

Since its release, PDTB has been applied to discourse parsing in [42, 22, 76]. The discourse parser described in this thesis follows the approach of [22] and is first described in [65].

2.2 Biomedical Discourse Relation Bank

Biomedical Discourse Relation Bank (BioDRB) [54] is a discourse relation corpus annotated over 24 open access full-text articles from the GENIA corpus [33] (biomedical domain).[1] The annotation methodology follows the Penn Discourse Treebank (PDTB) [52] framework, i.e. discourse relations are strictly binary information relations between eventualities or propositions mentioned in text. BioDRB inherits argument naming conventions from PDTB: *Arg1* and *Arg2*, which is syntactically bound to the connective. The released version of BioDRB does not contain annotation for *supplementary information* and *attribution.*

Similar to PDTB, BioDRB discourse connectives belong to a closed-class lexical items. However, unlike for PDTB with 3 well-defined syntactic classes of *subordinating conjunctions*, *coordinating conjunctions* and

[1] Unlike for PDTB, there is no reference tokenization or syntactic parse trees for BioDRB.

Type	**Count**	**%**
Explicit	2,636	45.0%
Implicit	3,001	51.2%
AltLex	193	3.3%
NoRel	29	0.5%
Total	5,859	100.0%

Table 2.3: The distribution of annotated relation types in BioDRB (from [54]).

discourse adverbials, BioDRB additionally covers *subordinators* (see Example 4). Most members of the discourse connective set are ambiguous with respect to discourse and non-discourse usage (e.g. *and*, that can also coordinate noun phrases, etc.).

> (4) Recent observations demonstrated that *IL-17 can also activate osteoclastic bone resorption* by **the induction of RANKL (receptor activator of nuclear factor κB [NF-κB] ligand), which is involved in bony erosion in RA** [7].
> (Purpose:Enablement)

Similar to PDTB, BioDRB annotates *explicit*, *implicit* and *alternatively lexicalized* (AltLex) discourse relations. However, *entity-based coherence relations* (EntRel), that in PDTB were not annotated for a relation sense, are eliminated as a separate type. Instead, the relation sense hierarchy was modified by additional relation senses – "Continuation" and "Background". The BioDRB relation sense hierarchy is addressed in the following subsection and Table 2.3 provides the distribution of the BioDRB relation types.

In comparison to PDTB, the ratio of implicit discourse relations is higher; however, this is mainly due to the changes to the EntRel type, since the relation is realized implicitly. For instance, relations with the sense "Background" are in 99% of instances realized implicitly and with the sense "Continuation" in 97% of instances.

2.2.1 BioDRB Discourse Relation Senses

The Table 2.4 presents the BioDRB discourse relation sense hierarchy. The main changes from PDTB are the following (from [54]):

- The four top-level senses of the PDTB hierarchy are removed; thus, the BioDRB sense hierarchy is two-level.

Type	**Sub-Type**
Cause	reason results claim justification
Condition	hypothetical factual non-factual
Purpose	goal enablement
Temporal	synchronous precedence succession
Concession	expectation contra-expectation
Alternative	chosen-alternative conjunctive disjunctive
Contrast	
Instantiation	
Conjunction	
Exception	
Similarity	
Continuation	
Circumstance	forward-circumstance backward-circumstance
Background	forward-background backward-background
Restatement	equivalence generalization specification
Reinforcement	

Table 2.4: BioDRB discourse relation sense hierarchy (from [54]).

PDTB Top-Level Sense (Class)	**BioDRB Top-Level Sense (Type)**
Comparison	Concession, Contrast
Contingency	Cause, Condition, Purpose
Temporal	Temporal
Expansion	Alternative, Background, Circumstance, Conjunction, Continuation, Exception, Instantiation, Reinforcement, Restatement, Similarity

Table 2.5: Mapping of BioDRB senses to PDTB 4 top-level senses (from [54]).

- The ‘Temporal’ top-level sense is retained as type.
- Some sub-type (third level) senses are combined together:
  - e.g. ‘present-factual’ and ‘past-factual’ are combined into ‘factual’.
- A new senses were introduced:
  - ‘Continuation’ and ’Background’ to cover EntRel relations.
  - ‘Purpose’, ‘Similarity’, and ‘Reinforcement’, which are either elaborations or re-names of PDTB senses.
- Separate *pragmatic* senses were eliminated, and became sub-types of ‘Cause’ relation – ’Claim’ and ’Justification’.

As a result of these changes, BioDRB has 16 top-level senses; however, they are still could be mapped to the PDTB four top-level senses (see Table 2.5 from [54]).

The further differences and similarities between the corpora are elaborated task-specifically in Chapter 4.

2.3 Italian LUNA Corpus

The Italian LUNA Corpus [15] is a collection of 723 human-machine (approximately 4,000 turns & 5 hours of speech) and 572 human-human (approximately 26,500 turns & 30 hours of speech) dialogs in the hardware/software help desk domain. The dialogs are conversations of the users

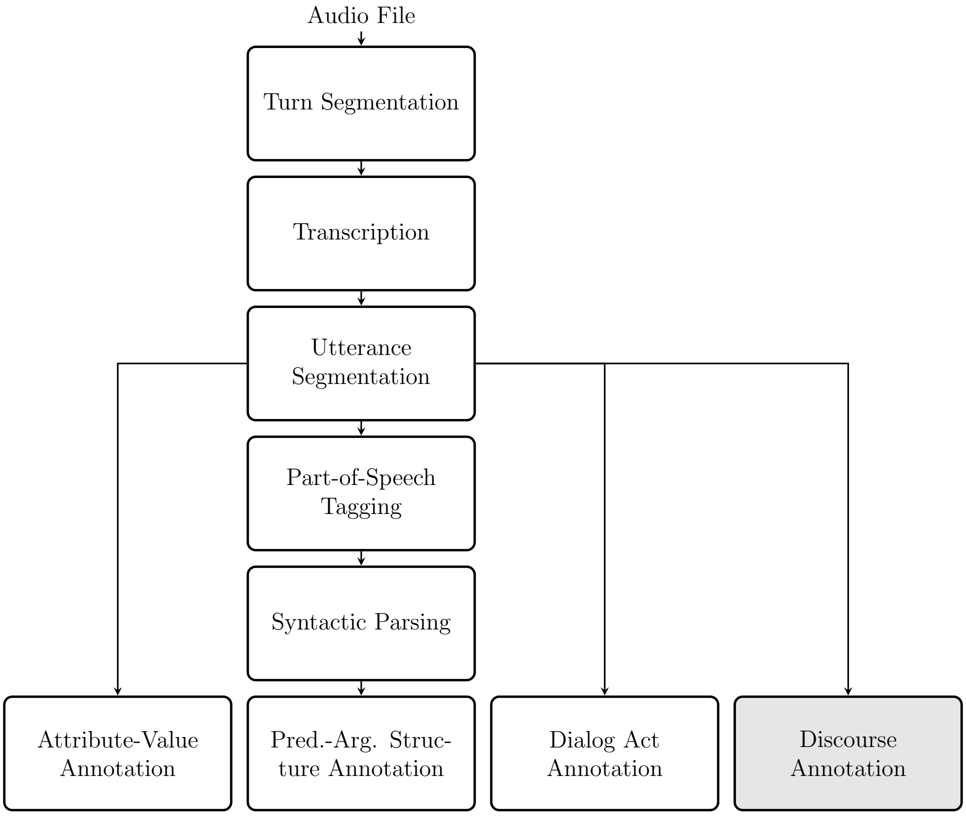


Figure 2.1: Italian LUNA Corpus annotation process (extended from [15]).

involved in problem solving. While human-human dialogs are recording of the real user-operator conversations, human-machine dialogs are collected using Wizard of Oz (WOZ) technique: the human agent (wizard) reacting to user requests is following one of the ten scenarios identified as most common by the help desk service provider. Text-to-Speech Synthesis (TTS) was used to provide responses to the users. The dialogs are organized in transcriptions and annotations defined within FP6 LUNA Project.

2.3.1 LUNA Annotation Protocol

The general process of multi-level annotation can be seen on Figure 2.1. After the dialogs were transcribed and split into utterances, they have been annotated at word-level. The word-level annotation consists of lemmas, part-of-speech tags and morpho-syntactic information following EAGLES corpora annotation [39]. Dialog act, attribute-value, and discourse an-

Annotation Level	**# of dialogs**	
	Hum-Mac	**Hum-Hum**
Attribute-Value	723	572
Dialog Act	224	81
Pred.-Arg. Structure (FrameNet)	129	78
Discourse Relation (PDTB)	–	60

Table 2.6: Statistics on LUNA Corpus annotation levels.

notation has been done on segmented dialogs at utterance level as well. However, predicate argument annotation requires POS-tagging and syntactic parsing. This was done semi-automatically using the Bikel parser trained on an Italian corpus [13] with subsequent manual correction. Different levels of annotation cover different subsets of the corpus. Table 2.6 provides information on the amount of dialogs annotated at each level.

In the following sub-section we briefly overview the other levels of annotation of the LUNA corpus, emphasizing the levels most relevant to the thesis –attribute-value and discourse relation annotation.

2.3.2 Attribute-Value Annotation

The attribute-value annotation of LUNA corpus uses a predefined ontology of concepts. There is an important distinction between the **attribute** of the concept, the **value** of the concept, and the **span** of the concept.

Since the domain of the LUNA corpus is hardware/software help desk, the concepts are sets of domain-specific entities such as *hardware*, *peripheral*, etc. and actions such as *hardware operation*, *network operation*, etc.. However, the ontology also contains concepts generic concepts such as *user*, *number*, *time*, etc..

The ontology consists of 45 unique concepts organized into two levels with the 26 top-level concepts. The second level of concepts can be seen as as properties of the top-level concept. For example, for the top-level 'generic' concept *user*, the second level concepts are *name*, *surname*, *position*, *data*, etc.; for the top-level concept *computer*, the second level concepts are *type* (e.g. PC or laptop) and *brand* (e.g. DELL or HP). For the purposes of Spoken Language Understanding the two levels are always considered together as an **attribute** of a concept. **Values** of concepts, on

```
ciao
<concept attribute="User.name" value="Paola">Paola</concept>
<filler/>
ho
<concept attribute="ProblemHardware.type" value="problem_generic">
un problema
</concept>
con la sta
<concept attribute="Peripheral.type" value="keyboard"/>
con la tastiera
</concept>
ho
<concept attribute="ProblemHardware.type" value="problem_hardware"/>
un tasto staccato
</concept>
mi è rimasto in mano e
<concept attribute="GenericAction.negate" value="non"/>non</concept>
<concept attribute="GenericAction.negate" value="non"/>non</concept>
<concept attribute="GenericAction.actionType" value="to_use"/>
riesco più a usarla
</concept>
```

Figure 2.2: LUNA attribute-value annotation example.

the other hand, are in the *computer.type* example are *PC* or *laptop*. The span of the concept is the portion of an utterance string – a number of tokens – covered by the concept.

To better illustrate the notions consider the utterance *ciao Paola ho un problema con la sta con la tastiera ho un tasto staccato mi è rimasto in mano e non non riesco più a usarla* (English: *Hi, Paola. I have a problem with the pri[nter] with the keyboard. I have a button off that remained in my hand ... cannot use it anymore.*), whose simplified attribute-value annotation is presented in Figure 2.2. In the example, the last concept has the **attribute** *GenericAction.actionType*, the **value** *to_use*, and its **span** covers four tokens: *riesco più a usarla*.

The attribute-value annotation level is used to train Spoken Language Understanding models. The goal of Chapters 5, 7, and 6 is to transfer this annotation across languages.

2.3.3 Dialog Act Annotation

The goal of dialog act annotation is to identify the function of an utterance within dialog that reflects speaker intentions. The LUNA Dialog act annotation was inspired by DAMSL [14], TRAINS [71], and DIT++ [8]. The most frequent dialog acts from these taxonomies are grouped into three [15]:

- *Core Dialog Acts* (8) are main actions in the dialog, such as *request of information*, *response*, or *performing the task.*
- *Conventional/Discourse Management Acts* (4) are utterances such as *greetings*, *apologies*, etc. whose function is to maintain general dialog cohesion.
- *Feedback/Grounding Acts* (3) are utterances whose function is to *acknowledge*, *provide feedback*, or just time fillers.

The unit of annotation for dialog acts is an utterance. However, due to the overlapping turns (both speakers speaking), an utterance can span several turns. Thus, the dialog act annotation was preceded by additional utterance segmentation.

2.3.4 Predicate Argument Annotation

The predicate argument annotation of LUNA Corpus is based on FrameNet model [1]. The FrameNet semantics cover a set of prototypical situations (*frames*) that involve certain number of entities playing a specific role in this situation (*frame elements*). A frame is triggered by a *target word or expression* from a set defined in the theory. All lexical units in this set have the same semantics. As it was already mentioned predicate argument annotation made use of syntactic parse trees.

2.3.5 LUNA Discourse Annotation

A set of 60 human-human dialogs from Italian LUNA Corpus was annotated with Penn Discourse Treebank-style discourse relations in [70]. The

Type	**Count**	**%**
Explicit	1,052	65.5%
Implicit	487	30.3%
AltLex	11	0.7%
EntRel	56	3.5%
Total	1,606	100.0%

Table 2.7: Distribution of LUNA discourse relation types (from [70])
.

annotation is carried on the raw text without considering other levels of annotation.

The argument selection procedure follows the PDTB in 'minimality principle', i.e. only the text string minimally necessary to interpret the relation is selected for each argument. As a difference from PDTB, the authors have lifted the adjacency principle for the annotation of implicit relations, since a dialog consists of interleaving speaker turns.

Additional difference is the re-definition of arguments of discourse relations. While in PDTB arguments are defined *syntactically*, in LUNA they are defined *semantically* [70]: in PDTB Argument 2 is the argument syntactically attached to a discourse connection; in LUNA, however, every argument has a sense specific semantic role regardless of its syntactic position in the relation. This is particularly the case for *causal* relation subtypes *reason* and *result*. This decision, however, have implications for discourse parsing utilizing syntactic features and processing Arguments 1 and 2 differently.

The Table 2.7 presents the distribution of Explicit, Implicit, AltLex and entity relations annotated in LUNA Corpus. In LUNA Corpus the ratio of Explicit relations is 65.5%, which is much higher that in PDTB (45.5%).

LUNA Discourse Relation Senses

The further adaptation of the PDTB framework to spontaneous dialogs consists of the revision of the relation senses (see Table 2.8 from [70]). While maintaining the four top-level relation sense classes, the second-level was revised by adding or removing some labels (see the Table). The major difference form PDTB with respect to the sense hierarchy is the

Class	Type	Sub-Type
Comparison	*Contrast*	semantic speech-act
	Concession	semantic propositional epistemic speech-act pragmatic
Contingency	*Goal*	semantic speech-act
	Cause	semantic epistemic speech-act
	Condition	semantic epistemic speech-act
Expansion	*Conjunction*	semantic speech-act
	Instantiation	
	Restatement	specification equivalence
	Exception	
	Alternative	semantic speech-act
Temporal	*Synchronous*	
	Asynchronous	

Table 2.8: LUNA discourse relation sense hierarchy (from [70]).

third level of senses.

The third level is modified to distinguish the intention of the speakers or an epistemic inferences (in PDTB they were considered as pragmatic). Since LUNA is a corpus of spoken dialogs the speaker's intentions and inferences are more important. Consequently, non-semantic interpretations of the connectives are further refined. In [70] the third level is replaced by this refined classification.

The discourse relation annotation of LUNA Corpus is used in Chapter 8 for Discourse Connective Detection in conversational data.

2.3.6 Anonymization

Within FP7 PortDial Project LUNA Corpus has gone through additional process of anonymization. Sensitive private information, such as personal names, phone numbers were replaced with random values: named entities were replaced with a random named entity of the same type drawn from a list of common Italian entities. and phone numbers were replaced with a random numeric sequences. A special attention was given to preserve the distribution of token frequencies within anonymized concept values.

Additionally, statistical Spoken Language Understanding (SLU) model was trained and tested on anonymized human-machine data to ensure that the step has no significant impact on the performance. The SLU model trained on original LUNA Corpus has Concept Error Rate (CER) of 21.5%, and the model trained and tested on anonymization corpus has CER of 21.7% (the difference is insignificant).

2.4 Multilingual LUNA Corpus[2]

The development of annotated corpora is a critical process in the development of speech applications for multiple target languages. While the technology to develop a monolingual speech application has reached satisfactory results (in terms of performance and effort), porting an existing application from a *source* language to a *target* language is still a very expensive task. In this section we describe Multilingual LUNA Corpus and address the challenges of the manual creation of multilingual corpora.

2.4.1 The Problem

Speech services are becoming increasingly spread (e.g. call centers, smartphones, etc.). The common limitation of the most available speech services is the lack of multilingual support: the services are developed only for the languages with rich available resources (usually English). Consequently, the large user bases of speakers of other languages are left out. The main

[2] The Section is partially published in E.A. Stepanov, G. Riccardi and A.O. Bayer. “The Development of the Multilingual LUNA Corpus for Spoken Language System Porting”, LREC, Reykjavik, 2014. [67]

reason for this is the fact that developing the same speech service in another language is an expensive manual effort; since it requires additional data collection and annotation. An alternative is an automatic cross-language porting of an existing service to another language via translation. However, it has severe data resource limitations. (1) Available multilingual resources such as aligned corpora are few in number and are different from conversational data in style. (2) Annotation in most monolingual language resources, such as Penn Treebank, is designed for linguistic analysis and hardly suitable for building data-driven spoken language systems. (3) Few existing parallel spoken conversation corpora represent resource-rich or close family language pairs.

There are very few parallel spoken conversation corpora specifically designed for building data-driven spoken language systems. The available ones are either translated to close languages (e.g. PORTMEDIA: French - Italian [41]), or from or to English (e.g. ATIS: English - Chinese [25]). Multilingual LUNA Corpus, on the other hand, is the translation of Italian LUNA Corpus via professional translation services that covers both close (Spanish), and distant family languages (Turkish and Greek). Thus, it allows for broader perspective on cross-language system portability. At the same time, it allows to address issues of cross-language porting differences to linguistic resource-rich and resource-poor languages. We first describe the source data – Italian LUNA corpus, and then specifics of the translation of conversation transcriptions.

2.4.2 Manual Creation of Multilingual Corpora: Professional Translation

Within the FP7 PortDial project, the Italian LUNA Human-Machine Corpus (all 723 dialogs) has been translated by expert translators to Spanish, Turkish and Greek. The translated corpus consists of text only (i.e. annotations have not been transferred); and is intended as a reference resource for research on data-driven spoken language system porting. In this section we describe the process and challenges associated with *manual* creation of multilingual conversational corpora.

IT	ES	TR	EN
ciao Paola	*hola, Paola,*	*merhaba Paola*	*hi Paola*
ho un problema	*tengo un problema*		*I have a problem*
con la sta *con la tastiera*	***con la pe...*** *on el teclado,*	*klavye ile* ***[“klavye ile”]***	***with the pri*** *with the keyboard*
		bir sornum var,	
ho un tasto staccato mi è rimasto in mano e	*tiene una tecla pegada, se me ha quedado en la mano y*	*bir tuş çıktı, elimde kaldı ve*	*I have a button off that remained in my hand and*
***non** non riesco più a usarla*	***no** no consigo usarla*	*artık kullanamıyorum* ***[“kullanamıyorum”]***	***cannot** use it anymore*

Table 2.9: An example of speech disfluency translations in a single utterance from Italian (IT) to Spanish (ES) and Turkish (TR). English translation (EN) is given for reference only.

Transcribed Speech Translation Artifacts

Since the LUNA Corpus is a corpus of transcribed speech, it is of a particular style: there is no sentence segmentation and punctuation. Additionally, it contains spontaneous speech artifacts such as speech disfluencies: repetitions, repairs, truncated words, etc., all of which have to be translated for a proper alignment to take place. Professional translators, on the other hand, are accustomed to working with written text. Thus, there are two translation artifacts: (1) punctuation is being inserted, which is a minor issue; and (2) speech disfluencies are *translated*, not recreated in the target language. Consequently, translation of spoken language phenomena have to be additionally inspected. Native speakers of target languages were queried for judgments on ‘naturalness’ of translated disfluencies and a policy was established for each language.

Speech Disfluency Translation Policy

The following policy was applied for speech disfluency translation. If the language pair is close enough to allow replicating disfluencies in the target language by the same morpho-syntactic means, without breaking the ‘naturalness’ of an utterance, they were replicated (Spanish, see example in Table 2.9). On the other hand, if the speech disfluency in target language requires different morpho-syntactic operation (e.g. determiner or preposition repetition in the source language is translated as a content word, postposition or suffix repetition), the disfluency is marked in text as such (Turkish, see example in Table 2.9). As a result, speech disfluencies are replicated in Spanish, and are marked in Turkish and Greek.

For example, in the utterance "*... ho un problema con la sta con la tastiera ... non non riesco più a usarla*" (English: 'I have a problem with the keyboard ... cannot use it anymore'), there are two speech disfluencies; and their translations are given in the Table 2.9. As example indicates, disfluencies are not easily replicable in every target language: e.g. for Turkish, because of word order differences and rich morphology, replication of the negation requires repetition of the whole verb, which was judged by native speakers to be 'unnatural'.

The translations of LUNA Corpus are aligned by dialog and utterance IDs; thus, the Multilingual LUNA Corpus constitutes a parallel Italian - Spanish - Turkish - Greek spoken dialog corpus readily available for translation and spoken dialog system research. The corpus provides aligned data for both close and distant family languages; thus, allows for broader perspective on cross-language system portability. The corpus is used in the later chapters for cross-language porting experiments.

2.5 Conclusion

The Chapter has provided descriptions of the corpora used for domain adaptation of Discourse Parsing and cross-language porting of Spoken Language Understanding tasks. In the following chapters we describe the tasks and additionally provide task-specific data analyses of the corpora.

Chapter 3

PDTB-Style Discourse Parsing[1]

Discourse relation parsing is an important task with the goal of understanding text beyond the sentence boundaries. One of the subtasks of discourse parsing is the extraction of argument spans of discourse relations. A relation can be either intra-sentential – to have both arguments in the same sentence – or inter-sentential – to have arguments span over different sentences. There are two approaches to the task. In the first approach the parser decision is not conditioned on whether the relation is intra- or inter-sentential. In the second approach relations are parsed separately for each class. In this chapter we evaluate the two approaches to argument span extraction on Penn Discourse Treebank explicit relations; and the problem is cast as token-level sequence labeling. We show that processing intra- and inter-sentential relations separately, reduces the task complexity and significantly outperforms the single model approach. The parser described in this Chapter is further developed in Chapter 4 for cross-domain robustness.

3.1 Introduction

Discourse analysis is one of the most challenging tasks in Natural Language Processing, that has applications in many language technology areas such as opinion mining, summarization, information extraction, etc. (see [73] and [69] for detailed review). With the availability of annotated corpora,

[1] The Chapter is published in E.A. Stepanov and G. Riccardi. "Comparative Evaluation of Argument Extraction Algorithms in Discourse Relation Parsing", *13th International Conference on Parsing Technologies (IWPT)*, Nara, Japan, 2013. [65]

such as Penn Discourse Treebank (PDTB) [52], statistical discourse parsers were developed [42, 22, 76].

PDTB adopts non-hierarchical binary view on discourse relations: Argument 1 (*Arg1*) and Argument 2 (*Arg2*), which is syntactically attached to a discourse connective. Thus, PDTB-based discourse parsing can be roughly partitioned into discourse relation detection, argument position classification, argument span extraction, and relation sense classification. For discourse relations signaled by a connective (explicit relations), discourse relation detection is cast as classification of connectives as discourse and non-discourse. Argument position classification involves detection of the location of *Arg1* with respect to *Arg2*: usually either the same sentence (SS) or previous ones (PS).[2] Argument span extraction, on the other hand, is extraction (labeling) of text segments that belong to each of the arguments. Finally, relation sense classification is the annotation of relations with the senses from PDTB.

Since arguments of explicit discourse relations can appear in the same sentence or in different ones (i.e. relations can be intra- or inter-sentential); there are two approaches to argument span extraction. In the first approach the parser decision is not conditioned on whether the relation is intra- or inter-sentential (e.g. [22]). In the second approach relations are parsed separately for each class (e.g. [42, 76]). In the former approach argument span extraction is applied right after discourse connective detection, while the latter approach also requires argument position classification.

The decision on argument span can be made on different levels: from token-level to sentence-level. In [22] the decision is made on token-level, and the problem is cast as sequence labeling using conditional random fields (CRFs) [38]. In this chapter we focus on argument span extraction, and extend the token-level sequence labeling approach of [22] with the separate models for arguments of intra-sentential and inter-sentential explicit discourse relations. To compare to the other approaches (i.e. [42] and [76]) we adopt the immediately previous sentence heuristic to select a candidate *Arg1* sentence for the inter-sentential relations. Additionally to the heuris-

[2] We use the term *inter-sentential* to refer to a set of relations that includes both previous sentence (*PS*) and following sentence (*FS*) *Arg1*. *Intra-sentential* and same sentence (*SS*) relations, on the other hand, are the same set.

tic, we train and test CRF argument span extraction models to extract *exact* argument spans.

The chapter is structured as follows. In Section 3.2 we briefly present the information about the Penn Discourse Treebank corpus that is relevant to the experiments. Section 3.3 describes related works. Section 3.4 defines the problem and assesses its complexity. In Section 3.5 we describe argument span extraction cast as the token-level sequence labeling; and in Section 3.6 we present the evaluation of the two approaches – either single or separate processing of intra- and inter-sentential relations – on PDTB explicit relations. Section 3.7 provides concluding remarks.

3.2 The Penn Discourse Treebank

The Penn Discourse Treebank (PDTB) [52] is a corpus that contains discourse relation annotation on top of WSJ corpus; and it is aligned with Penn Treebank (PTB) syntactic tree annotation. Discourse relations in PDTB are binary: *Arg1* and *Arg2*, where *Arg2* is an argument syntactically attached to a discourse connective. With respect to *Arg2*, *Arg1* can appear in the same sentence (SS case), one of the preceding (PS case) or following (FS case) sentences.

A discourse connective is a member of a well defined list of 100 connectives and a relation expressed via such connective is an *Explicit* relation. Discourse relations are annotated using 3-level hierarchy of senses. The top level (level 1) senses are the most general: *Comparison*, *Contingency*, *Expansion*, and *Temporal* [52]. There are other types of discourse and non-discourse relations annotated in PDTB; however, they are out of the scope of this chapter.

3.3 Related Works

In this Section we briefly describe published works on the PDTB-style discourse parsing subtasks: discourse connective detection, relation sense classification, argument position classification,and argument span extraction.

The subtask of discourse connective detection was addressed in [51]. The authors applied machine learning methods using lexical and syntactic features and achieved high classification performance (F_1: 94.19%, 10 fold cross-validation on PDTB sections 02-22). Later, these results were further improved with additional lexico-syntactic and path features in [42] (F_1: 95.76%).

After a discourse connective is identified as such, it is classified into relation senses annotated in PDTB. [51] classify discourse connectives into 4 top level senses – *Comparison*, *Contingency*, *Expansion*, and *Temporal* – and achieve accuracy of 94.15%, which is slightly above the inter-annotator agreement. In this chapter we focus on the parsing steps after discourse connective detection; thus, we use gold reference connectives and their senses as features.

The approaches using only the argument position classification even though useful are incomplete; as they do not make decision on argument spans. [74] and [18], following them, used machine learning methods to identify head words of the arguments of explicit relations expressed by discourse connectives. Prasad et al [53], on the other hand, addressed a more difficult task of identification of sentences that contain *Arg1* for cases when arguments are located in different sentences.

[16] and [42] approach the problem of argument span extraction on syntactic tree node-level. In the former, it is a rule based system that covers limited set of connectives; whereas in the latter it is a machine learning approach with full PDTB coverage. Both apply syntactic tree subtraction to get argument spans. [76] approach the problem on a constituent-level: authors first decide whether a constituent is a valid argument and then whether it is *Arg1*, *Arg2*, or neither. [22] (and further [23, 24]), on the other hand, cast the problem as token-level sequence labeling. In this work we follows the approach of [22].

3.4 Problem Definition

In the introduction we mentioned Immediately Previous Sentence Heuristic for *Arg1* of inter-sentential explicit relations and Argument Position Classi-

	SingFull	SingPart	MultFull	MultPart	Total
ARG1					
IPS	**3,192** (44.2%)	**1,880** (26.0%)	370 (5.1%)	107 (1.5%)	5,549 (76.8%)
NAPS	993 (13.8%)	551 (7.6%)	71 (1.0%)	51 (0.7%)	1,666 (23.1%)
FS	2 (0.0%)	0 (0.0%)	1 (0.0%)	5 (0.0%)	8 (0.1%)
Total	4,187 (58.0%)	2,431 (33.7%)	442 (6.1%)	163 (2.3%)	7,223 (100%)
ARG2					
SS/Total	5,181 (71.7%)	1,936 (26.8%)	84 (1.2%)	22 (0.3%)	7,223 (100%)

Table 3.1: Distribution of *Arg1* with respect to the location (rows) and extent (columns) (partially copied from [52]); and distribution of *Arg2* with respect to extent **in inter-sentential explicit discourse relations**. SS = same sentence as the connective; IPS = immediately previous sentence; NAPS = non-adjacent previous sentence; FS = some sentence following the sentence containing the connective; SingFull = Single Full sentence; SingPart = Part of single sentence; MultFull = Multiple full sentences; MultPart = Parts of multiple sentences.

fication as a prerequisite for processing intra- and inter-sentential relations separately. In this section we analyze PDTB to assess the complexity and potential accuracy of the heuristic and the classification task.

3.4.1 Immediately Previous Sentence Heuristic

According to [52]'s analysis of explicit discourse relations annotated in PDTB, out of 18,459 relations, 11,236 (60.9%) have both of the arguments in the same sentence (SS case), 7,215 (39.1%) have *Arg1* in the sentences preceding the *Arg2* (PS case), and only 8 instances have *Arg1* in the sentences following *Arg2* (FS case). Since FS case has too few instances it is usually ignored. For the PS case, the *Arg1* is located either in Immediately Previous Sentences (IPS: 30.1%) or in some Non-Adjacent Previous Sentences (NAPS: 9.0%).

CRF-based discourse parser of [22], which processes SS and PS cases with the same model, uses ±2 sentence window as a hypothesis space (5 sentences: 1 sentence containing the connective, 2 preceding and 2 following sentences). The window size is motivated by the observation that it entirely covers arguments of 94% of all explicit relations. The authors also report that the performance of the parser on inter-sentential relations (i.e. mainly PS case) has F-measure of 36.0. However, since in 44.2% of inter-sentential explicit discourse relations *Arg1* fully covers the sentence immediately preceding *Arg2* (see Table 3.1 partially copied from [52]), the heuristic that selects the immediately previous sentence and tags all of its tokens as *Arg1* already yields F-measure of 44.2 over all PDTB (the performance on the test set may vary).

The same heuristic is mentioned in [42] and [76] as a majority classifier for the relations with *Arg1* in previous sentences.

Compared to the ±2 window, the heuristic covers *Arg1* of only 88.4% explicit discourse relations (60.9% SS + 27.5% PS); since it ignores all the relations with *Arg1* in Non-Adjacent Previous Sentences (NAPS) (9.0% of all explicit relations), and does not accommodate *Arg1* spanning multiple immediately preceding sentences (2.6% of all explicit relations). Nevertheless, 70.2% of all PS explicit relations have *Arg1* entirely inside the immediately previous sentence. Thus, the integration of the heuristic is expected to improve the argument span extraction performance for inter-sentential *Arg1*.

In 98.5% of all PS cases *Arg2* is within the sentence containing the connective (remaining 1.5% are multi-sentence *Arg2*); and in 71.7% of all PS cases it fully covers the sentence containing the discourse connective (see Table 3.1). Thus, similar heuristic for *Arg2* is to tag all the tokens of the sentence except the connective as *Arg2*.

For the heuristics to be applicable, a discourse connective has to be classified as requiring its *Arg1* in the same sentence (SS) or the previous ones (PS), i.e. it requires argument position classification.

3.4.2 Argument Position Classification

Explicit discourse connectives, annotated in PDTB, belong to one of the three syntactic categories: (1) subordinating conjunctions (e.g. *when*), (2) coordinating conjunctions (e.g. *and*), and (3) discourse adverbials (e.g. *for example*). With few exceptions, a discourse connective belongs to a single syntactic category (see Appendix A in [35]). Each of these syntactic categories has a strong preference on the position of *Arg1*, depending on whether the connective appears sentence-initially or sentence-medially. Here, a connective is considered sentence-initial if it appears as the first

	Sentence Initial		Sentence Medial	
	SS	PS	SS	PS
Coordinating	10 (0.05%)	2,869 (15.54%)	3,841 (20.81%)	202 (1.09%)
Subordinating	1,402 (7.60%)	114 (0.62%)	5,465 (29.61%)	83 (0.45%)
Discourse Adverbial	13 (0.07%)	1,632 (8.84%)	495 (2.68%)	2,325 (12.60%)

Table 3.2: Distribution of discourse connectives in PDTB with respect to syntactic category (rows) and position in the sentence (columns) and the location of *Arg1* as in the same sentence (SS) as the connective or the previous sentences (PS). The case when *Arg1* appears in some following sentence (FS) is ignored, since it has only 8 instances.

sequence of words in a sentence. Table 3.2 presents the distribution of discourse connectives in PDTB with respect to the syntactic categories, their position in the sentence, and having *Arg1* in the same or previous sentences. The distribution of sentence-medial discourse adverbials, which is the most ambiguous class, between SS and PS cases is 17.5% to 82.5%; for all other classes it higher than 90% to 10%. Thus, the overall accuracy of the SS vs. PS majority classification using just syntactic category and position information is already 95.0%.

When analyzed on per connective basis, the observation is that some connectives require *Arg1* in the same or previous sentence irrespective of their position in the sentence. For instance, sentence-initial subordinating conjunction *so* always has its *Arg1* in the previous sentence; and the parallel sentence-initial subordinating conjunction *if..then* in the same sentence. Others, such as sentence-medial adverbials *however* and *meanwhile* mainly require their *Arg1* in the previous sentence. Even though low, there is still an ambiguity: e.g. for sentence-medial adverbials *also*, *therefore*, *still*, *instead*, *in fact*, etc. *Arg1* appears in SS and PS cases evenly. Consequently, assigning the position of the *Arg1* considering the discourse connective, together with its syntactic category and its position in the sentence, for PDTB will be correct in more than 95% of instances.

In the literature, the task of argument position classification was addressed by several researchers (e.g. [53], [42]). [42], for instance, report F_1 of 97.94% for a classifier trained on PDTB sections 02-21, and tested on section 23. The task has a very high baseline and even higher performance on supervised machine learning, which is an additional motivation to process intra- and inter-sentential relations separately.

Feature	**ABBR**	**Arg2**	**Arg1**
Token	TOK	Y	Y
POS-Tag	POS		
Lemma	LEM	Y	Y
Inflection	INFL	Y	Y
IOB-Chain	IOB	Y	Y
Connective Sense	CONN	Y	Y
Boolean Main Verb	BMV		Y
Arg2 Label	ARG2		Y

Table 3.3: Feature sets for *Arg2* and *Arg1* argument span extraction in [22]

3.5 Parsing Models

We replicate and evaluate the discourse parser of [22], then modify it to process intra- and inter-sentential explicit relations separately. This is achieved by integrating Argument Position Classification and Immediately Previous Sentence heuristic into the parsing pipe-line.

Since the features used to train argument span extraction models for both approaches are the same, we first describe them in Subsection 3.5.1. Then we proceed with the description of the single model discourse parser (our baseline) and separate models discourse parser, Subsections 3.5.2 and 3.5.3, respectively.

3.5.1 Features

The features used to train the models for *Arg1* and *Arg2* are given in Table 3.3. Besides the token itself (*TOK*), the rest of the features is described below.

- *Lemma* (*LEM*) and *inflectional* affixes (*INFL*) are extracted using *morpha* tool [46], that requires token and its POS-tag as input. For instance, for the word *flashed* the lemma and infection features are '*flash*' and '*+ed*', respectively.

- *IOB-Chain* (*IOB*) is the path string of the syntactic tree nodes from the root node to the token, prefixed with the information whether a token is at the beginning (B-) or inside (I-) the constituent. The feature is extracted using the *chunklink* tool [7]. For example, the

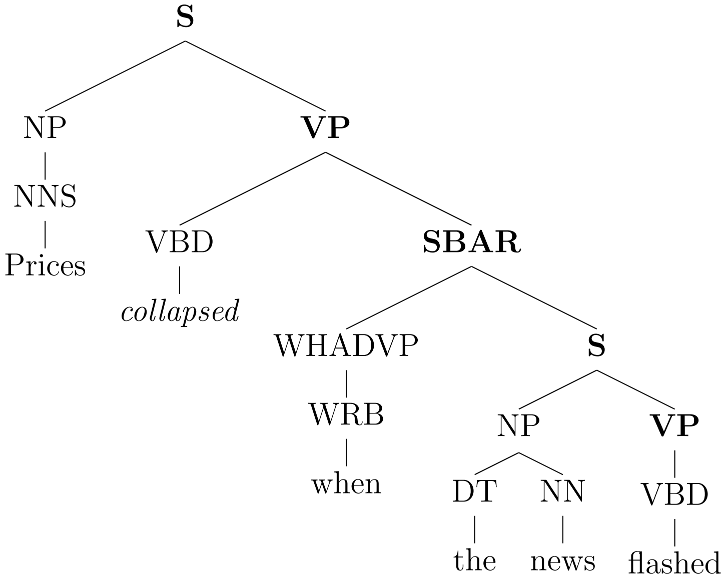


Figure 3.1: Example syntactic parse tree with the path string of the nodes for *flashed* in bold.

IOB-Chain '*I-S/B-VP*' indicates that a token is the first word of the verb phrase (*B-VP*) of the main clause (*I-S*).

- *PDTB Level 1 Connective sense* (*CONN*) is the most general sense of a connective in PDTB sense hierarchy: one of *Comparison*, *Contingency*, *Expansion*, or *Temporal*. For instance, a discourse connective *when* might have the CONN feature '*Temporal*' or '*Contingency*' depending on the discourse relation it appears in, or '*NULL*' in case of non-discourse usage. The value of the feature is '*NULL*' for all tokens except the discourse connective.

- *Boolean Main Verb* (*BMV*) is a feature that indicates whether a token is a main verb of a sentence or not [77]. For instance in the sentence *Prices collapsed when the news flashed*, the main verb is *collapsed*; thus, its BMV feature is '1', whereas for the rest of tokens it is '0'.

- *Previous Sentence Feature* (*PREV*) signals if a sentence immediately precedes the sentence starting with a connective, and its value is the first token of the connective [22]. For instance, if some sentence *A* is followed by a sentence *B* starting with discourse connective *On the other hand*, all the tokens of the sentence *A* have the *PREV* feature

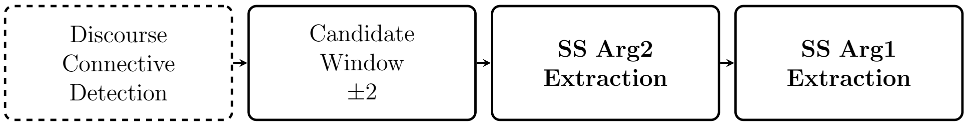


Figure 3.2: Single model discourse parser architecture of [22]. CRF argument span extraction models are in bold.

value '*On*'. The feature is similar to a heuristic to select the sentence immediately preceding a sentence starting with a connective as a candidate for *Arg1*.

- *Arg2 Label* (*ARG2*) is an output of *Arg2* span extraction model, and it is used as a feature for *Arg1* span extraction. Since for sequence labeling we use IOBE (Inside, Out, Begin, End) notation, the possible values of *ARG2* are IOBE-tagged labels, i.e. '*ARG2-B*' – if a word is the first word of *Arg2*, '*ARG2-I*' – if a word is inside the argument span, '*ARG2-E*' – if a word is in the last word of *Arg2*, and '*O*' otherwise.

CRF++[3] – conditional random field implementation we use – allows definition of feature templates. Via templates these features are enriched with n-grams: tokens with 2-grams in the window of ±1 tokens, and the rest of the features with 2 & 3-grams in the window of ±2 tokens.

For instance, labeling a token as *Arg2* is an assignment of one of the four possible labels: ARG2-B, ARG2-I, ARG2-E and O (ARG2 with IOBE notation). The feature set (token, lemma, inflection, IOB-chain and connective sense (see Table 3.3)) is expanded by CRF++ via template into 55 features ($5 * 5$ unigrams, 2 token bigrams, $4 * 4$ bigrams and $4 * 3$ trigrams of other features).

3.5.2 Single Model Discourse Parser

The discourse parser of [22] is a cascade of CRF models to sequentially label *Arg2* and *Arg1* spans (since *Arg2* label is a feature for *Arg1* model) (see Figure 3.2). There is no distinction between intra- and inter-sentential relations, rather the single model jointly decides on the position and the

[3] https://code.google.com/p/crfpp/

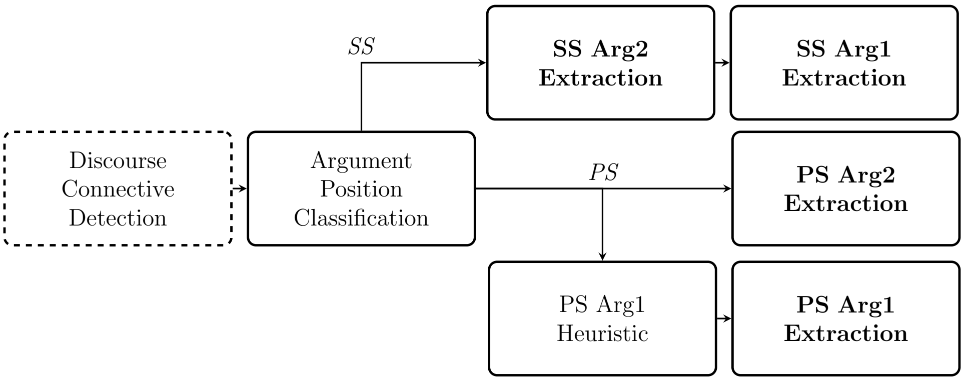


Figure 3.3: Separate models discourse parsing architecture. CRF argument span extraction models are in bold.

span of an argument (either *Arg1* or *Arg2*, not both together) in the window of ±2 sentences (the parser will be further abbreviated as *W5P* – Window 5 Parser).

The single model parser achieves F-measure of 81.7 for *Arg2* and 60.3 for *Arg1* using CONNL evaluation script. The performance is higher than [22] – *Arg2*: F_1 of 79.1 and *Arg1*: F_1 of 57.3 – due to improvements in feature and instance extraction, such as the treatment of multi-word connectives. These models are the baseline for comparison with separate models architecture. However, we change the evaluation method (see Section 3.6).

3.5.3 Separate Models Discourse Parser

Figure 3.3 depicts the architecture of the discourse parser processing intra- and inter-sentential relations separately. It is a combination of argument position classification with specific CRF models for each of the arguments of SS and PS cases, i.e. there are 4 CRF models – SS *Arg1* and *Arg2*, and PS *Arg1* and *Arg2* (following sentence case (FS) is ignored). SS models are applied in a cascade and, similar to the baseline single model parser, *Arg2* label is a feature for *Arg1* span extraction. These SS models are trained using the same feature set, with the exception of *PREV* feature: since we consider only the sentence containing the connective, it naturally falls out.

For the PS case, we apply a heuristic to select candidate sentences. Based on the observation that in PDTB for the PS case *Arg2* span is fully located in the sentence containing the connective in 98.5% of instances;

1. Classify connective as SS or PS;
2. If classified as SS:
    (a) Use SS Arg2 CRF model to label the sentence tokens for *Arg2*;
    (b) Use SS Arg1 CRF model to label the sentence tokens for *Arg1* using *Arg2* label as a feature;
3. If classified as PS
    (a) Select the sentence containing the connective and use PS Arg2 CRF model to label *Arg2* span;
    (b) Select the sentence immediately preceding the *Arg2* sentence and use PS Arg1 CRF model to label *Arg1* span.

Figure 3.4: The Process of Argument Span Extraction.

and *Arg1* span is fully located in the sentence immediately preceding *Arg2* in 71.7% of instances; we select sentences in these positions to train and test respective CRF models. The feature set for *Arg2* remains the same, whereas, from *Arg1* feature set we remove *PREV* and *Arg2* label (since in PS case *Arg2* is in different sentence, the feature will always have the same value of 'O').

For *Argument Position Classification* we train unigram BoosTexter [61] model with 100 iterations[4] on PDTB sections 02-22 and test on sections 23-24; and, similar to other researchers, achieve high results: $F_1 = 98.12$. The features are connective surface string, POS-tags, and IOB-chains. The results obtained using automatic features ($F_1 = 97.87$) are insignificantly lower (McNemar's $\chi^2(1, 1595) = 0.75, p = 0.05$); thus, this step will not cause deterioration in performance with automatic features. Here we used Stanford Parser [34] to obtain POS-tags and automatic constituency-based parse trees.

Since both argument span extraction approaches are equally affected by the discourse connective detection step, we use gold reference connectives. As an alternative, discourse connectives can be detected with high accuracy using addDiscourse tool [51]. In the separate models discourse parser, the

[4] The choice is based on the number of discourse connectives defined in PDTB.

steps of the process to extract argument spans given a discourse connective are given in Figure 3.4. The separate model parser with CRF models will be further abbreviated as *SMP*; and with the heuristics for PS case as *hSMP*.

3.6 Experiments and Results

We first describe the evaluation methodology. Then present evaluation of PS case CRF models against the heuristic. In subsection 3.6.3 we compare the performance of the single and separate model parsers on SS and PS cases of the test set separately and together. Finally, we compare the results of the separate model parser to [42] and [76].

3.6.1 Evaluation

There are two important aspects regarding the evaluation. First, in this work it is different from [22]; thus, we first describe it and evaluate the difference. Second, in order to compare the baseline single and separate model parsers, the error from argument position classification has to be propagated for the latter one; and the process is described in 3.6.1.

Since both versions of the parser are affected by automatic features, the evaluation is on gold features only. The exception is for *Arg2* label; since it is generated within the segment of the pipeline we are interested in. Unless stated otherwise, all the results for *Arg1* are reported for automatic *Arg2* labels as a feature. Following [22] PDTB is split as Sections 02-22 for training, 00-01 for development, and 23-24 for testing.

CONLL vs. String-based Evaluation

[22] report using CONLL-based evaluation script. However, it is not well suited for the evaluation of argument spans because the unit of evaluation is a chunk – a segment delimited by any out-of-chunk token or a sentence boundary. However, in PDTB arguments can (1) span over several sentences, (2) be non-contiguous in the same sentence. Thus, CONLL-based evaluation yields incorrect number of test instances: [22] report 1,028 SS

	Arg2	**Arg1**
CONNL-based	81.72	60.33
String-based	77.79	55.33

Table 3.4: Performance of the baseline ±2 window parser: CONNL-based vs. string-based evaluation reported as F_1. *Arg1* results are for gold *Arg2* labels.

and 617 PS test instances for PDTB sections 23-24 (see caption of Table 7 in the original paper), which is 1,645 in total; whereas there is only 1,595 explicit relations in these sections.

In our case, the evaluation is string-based; i.e. an argument span is correct, if it matches the whole reference string. Following [22] and [42], argument initial and final punctuation marks are removed; and precision (p), recall (r) and F_1 score are computed using the equations 3.1 – 3.3.

$$p = \frac{\text{Exact Match}}{\text{Exact Match} + \text{No Match}} \tag{3.1}$$

$$r = \frac{\text{Exact Match}}{\text{References in Gold}} \tag{3.2}$$

$$F_1 = \frac{2 * p * r}{p + r} \tag{3.3}$$

In the equations, *Exact Match* is the count of correctly tagged argument spans; *No Match* is the count of argument spans that do not match the reference string exactly (even one token difference is counted as an error); and *References in Gold* is the total number of arguments in the reference.

String-based evaluation of the single model discourse parser with gold features reduces F_1 for *Arg2* from 81.7 to 77.8 and for *Arg1* from 60.33 to 55.33 (see Table 3.4).

Error Propagation

Since the single model parser applies argument span extraction right after discourse connective detection, whereas in the separate model parser there is an additional step of argument position classification; for the two to be comparable an error from the argument position classification is propagated. Even though, the performance of the classifier is very high (98.12%)

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
hSMP	74.19	74.19	74.19	39.19	39.19	39.19
SMP	78.61	78.23	78.42	46.81	37.90	41.89

Table 3.5: Argument span extraction performance of the heuristics (*hSMP*) and the CRF models (*SMP*) on inter-sentential relations (PS case). Results are reported as precision (**P**), recall (**R**) and F-measure (**F1**)

there are still some misclassified instances. These instances are propagated to the counts of *Exact Match* and *No Match* of the argument span extraction. For example, if the argument position classifier misclassified an SS connective as PS; in the SS evaluation its *Arg1* and *Arg2* are considered as not recalled regardless of argument span extractor's decision (i.e. neither *Exact Match* nor *No Match*); and in the PS evaluation, they are both considered as *No Match*.

The separate model discourse parser results are reported without error propagation for in-class comparison of the heuristic and CRF models, and with error propagation for cross-class comparison with the single model parser.

3.6.2 Heuristic vs. CRF Models

The goal of this section is to assess the benefit of training CRF models for the extraction of exact argument spans of PS *Arg1* and *Arg2* on top of the heuristics. The performance of the heuristics (immediately previous sentence for *Arg1* and the full sentence except the connective for *Arg2*) and the CRF models is reported in Table 3.5. CRF models perform significantly better for *Arg2* (McNemar's $\chi^2(1, 620) = 7.48$, $p = 0.05$). Even though, they perform 2.7% better for *Arg1*, the difference is insignificant (McNemar's $\chi^2(1, 620) = 0.66$, $p = 0.05$). For both arguments, the CRF model results are lower than expected.

3.6.3 Single vs. Separate Models

To compare the single and the separate model parsers, the results of the former must be split into SS and PS cases. For the latter, on the other

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
W5P	87.57	84.51	86.01	71.73	62.97	67.07
SMP	90.36	87.49	88.90	70.27	66.67	68.42

Table 3.6: Performance of the single ±2 window (*W5P*) and separate model (*SMP*) parsers on argument span extraction of SS relations; reported as precision (**P**), recall (**R**) and F-measure (**F1**). For the *SMP* results are with error propagation from argument position classification.

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
W5P	71.12	59.19	64.61	40.06	22.74	29.01
hSMP	74.67	72.23	73.94	38.98	38.23	38.60
SMP	79.01	77.10	78.04	46.23	36.61	40.86

Table 3.7: Performance of the single model parser (*W5P*) and the separate model parser with the heuristics (*hSMP*) and CRF models (*SMP*) on argument span extraction of PS relations; reported as precision (**P**), recall (**R**) and F-measure (**F1**). For the separate model parsers, results include error propagation from argument position classification.

hand, we propagate error from the argument position classification step. For the PS case we also report the performance of the heuristic with error propagation.

Table 3.6 reports the results for the SS case, and Table 3.7 reports the results for the PS case. In both cases the separate model parser with error propagation from argument position classification step significantly outperforms the single model parser.

The performance of the separate model parsers (reported in Table 3.8) with heuristics and CRF models on all relations (SS + PS) both are significantly better than the performance of single ±2 window model parser (for *SMP* McNemar's $\chi^2(1, 1595) = 17.75$ for *Arg2* and $\chi^2(1, 1595) = 19.82$ for *Arg1*, $p = 0.05$).

3.6.4 Comparison of Separate Model Parser to State-of-the-Art

The separate model parser allows to compare argument span extraction cast as token-level sequence labeling to the syntactic tree-node level classification approach of [42] and constituent-level classification approach of [76]; since now the complexity and the hypothesis spaces are equal. For

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
W5P	81.47	74.42	77.79	61.90	46.96	53.40
hSMP	84.21	81.94	83.06	57.86	55.61	56.71
SMP	85.93	83.45	84.67	61.94	54.98	58.25

Table 3.8: Performance of the single model parser (*W5P*) and the separate model parser with the heuristics (*hSMP*) and CRF models (*SMP*) on argument span extraction of all relations; reported as precision (**P**), recall (**R**) and F-measure (**F1**). For the separate model parsers, results include error propagation from argument position classification.

	Arg2	**Arg1**
Lin et al. (2012)	82.23	59.15
Xu et al. (2012)	81.00	**60.69**
hSMP	80.04	54.37
SMP	**82.35**	57.26

Table 3.9: Comparison of the separate model parsers (with heuristics (*hSMP*) and CRFs (*SMP*)) to [42] and [76] reported as F-measure (**F1**). Trained on PDTB sections 02-21, tested on 23.

this purpose we train models on sections 02-21 and test on 23.

Unfortunately, the authors do not report the results on SS and PS cases separately, but only the combined results that include the heuristic. Moreover, the performance of the heuristic is mentioned to be 76.9% instead of 44.2% for the exact match (see IPS x SingFull cell in Table 3.1 or Table 1 in [52]). Thus, the comparison provided here is not definite. Since all systems have different components up the pipe-line, the only possible comparison is without error propagation. From the results in Table 3.9, we can observe that all the systems perform well on *Arg2*. As expected, for the harder case of *Arg1*, performances are lower.

3.7 Conclusion

In this chapter we compare two strategies for the argument span extraction: to process intra- and inter-sentential explicit relations by a single model, or separate ones. We extend the approach of [22] to argument span extraction cast as token-level sequence labeling using CRFs and integrate argument position classification and immediately previous sentence heuristic. The evaluation of parsing strategies on the PDTB explicit discourse relations shows that the models trained specifically for intra- and inter-sentential relations significantly outperform the single ±2 window models.

Chapter 4

Cross-Domain Discourse Parsing[1]

In Chapter 3 we have presented PDTB-Style Discourse Parsing cast as token-level sequence labeling. In this Chapter, on the other hand, we approach the problem of Cross-Domain Discourse Parsing. The parser developed in the previous chapter is applied to biomedical domain. The biomedical domain is of particular interest due to the availability of Biomedical Discourse Relation Bank (BioDRB). In the literature it was shown that the discourse parsing subtasks of discourse connective detection and relation sense classification do not generalize well across domains. In this chapter we evaluate feature-level domain adaptation techniques on the argument span extraction subtask. We demonstrate that the subtask generalizes well across domains.

4.1 Introduction

The release of the large discourse relation annotated corpora, such as Penn Discourse Treebank (PDTB) [52], marked the development of statistical discourse parsers [42, 22, 76, 65]. Recently, PDTB-style discourse annotation was applied to biomedical domain and Biomedical Discourse Relation Bank (BioDRB) [54] was released. This milestone marks the beginning of the research on cross-domain evaluation and domain adaptation of PDTB-style discourse parsers. In this chapter we address the question of how well

[1] The Chapter is published in E.A. Stepanov and G. Riccardi. "Towards Cross-Domain PDTB-Style Discourse Parsing", *EACL Workshops: The Fifth International Workshop on Health Text Mining and Information Analysis (Louhi)*, Gothenburg, Sweden, 2014. [66]

PDTB-trained discourse parser (news-wire domain) can extract argument spans of *explicit* discourse relations in BioDRB (biomedical domain).

The use cases of discourse parsing in biomedical domain are discussed in detail in [54]. Here, on the other hand, we provide very general connection between the two. The goal of Biomedical Text Mining (BioNLP) is to retrieve and organize biomedical knowledge from scientific publications; and detecting discourse relations such as, for instance, contrast and causality is an important step towards this goal [54]. To illustrate this point consider a quote from [6], given below.

> *The addition of an anti-Oct2 antibody did not interfere with complex formation (Figure 3, lane 6)*, since **HeLa cells do not express Oct2**. (Cause:Reason)

In the example, discourse connective since signals a causal relation between the clauses it connects. That is, the reason why '*the addition of an anti-Oct2 antibody did not interfere with complex formation*' is '*HeLa cells' not expressing Oct2*'.

As it was mentioned in Chapter 3, PDTB adopts non-hierarchical binary view on discourse relations: Argument 1 (*Arg1*) (in italics in the example) and Argument 2 (*Arg2*), which is syntactically attached to a discourse connective (in bold). Thus, a discourse relation is a triplet of a connective and its two arguments. In the literature [42, 65] PDTB-style discourse parsing is partitioned into discourse relation detection, argument position classification, argument span extraction, and relation sense classification. For the *explicit* discourse relations (i.e. signaled by a connective), discourse relation detection is cast as classification of connectives as discourse and non-discourse. Argument position classification, on the other hand, involves detection of the location of *Arg1* with respect to *Arg2*, that is to detect whether a relation is inter- or intra- sentential. Argument span extraction is the extraction (labeling) of text segments that belong to each of the arguments. Finally, relation sense classification is the annotation of relations with the senses from the sense hierarchy (PDTB or BioDRB).

To the best of our knowledge, the only subtasks that were addressed cross-domain are the detection of explicit discourse connectives [56, 55, 19] and relation sense classification [54]. While the discourse parser of [19][2] provides models for both domains and does identification of argument head words in the style of [74]; there is no decision made on arguments spans. Moreover, there is no cross-domain evaluation available for each of the models. In this chapter we address the task of cross-domain argument span extraction of *explicit* discourse relations. Additionally, we provide evaluation for cross-domain argument position classification as far as the data allows; since BioDRB lacks manual sentence segmentation.

The chapter is structured as follows. In Section 4.2 we present the comparative analysis of PDTB and BioDRB corpora and the relevant works on cross-domain discourse parsing. In Section 4.3 we present the experimental results. Section 4.4 provides concluding remarks.

4.2 PDTB vs. BioDRB Corpora Analysis and Related Cross-Domain Works

The two corpora used in our experiments are Penn Discourse Treebank (PDTB) [52] and Biomedical Discourse Relation Bank (BioDRB) [54]. Both corpora follow the same discourse relation annotation style over different domain corpora: PDTB is annotated on top of *Wall Street Journal* (WSJ) corpus (financial news-wire domain); and it is aligned with Penn Treebank (PTB) syntactic tree annotation; BioDRB, on the other hand, is a corpus annotated over 24 open access full-text articles from the GENIA corpus [33] (biomedical domain), and, unlike PDTB, there is no reference tokenization or syntactic parse trees. Here, on the other hand, we focus on the corpus differences relevant for discourse parsing tasks and cross-domain application of discourse parsing subtasks.

Discourse relations in both corpora are binary: *Arg1* and *Arg2*, where *Arg2* is an argument syntactically attached to a discourse connective. With respect to *Arg2*, *Arg1* can appear in the same sentence (SS case), one or several of the preceding (PS case) or following (FS case) sentences. A discourse connective is a member of a well defined list of connectives and a relation expressed via such connective is an *Explicit* relation. There are

[2] Made available on `https://code.google.com/p/discourse-parser/`

	PDTB	**BioDRB**
N. of Disc. Connectives	18,459	2,636
N. of Disc. Connective Types	100	123

Table 4.1: Differences between PDTB and BioDRB with respect to discourse connectives.

other types of discourse and non-discourse relations annotated in the corpora; however, they are out of the scope of this work. Discourse relations are annotated using a hierarchy of senses: even though the organization of senses and the number of levels are different between corpora, the most general top level senses are mapped to the PDTB top level senses: *Comparison*, *Contingency*, *Expansion*, and *Temporal* [54].

The difference between the two corpora with respect to discourse connectives is that in case of PDTB the annotated connectives belong to one of the three syntactic classes: subordinating conjunctions (e.g. *because*), coordinating conjunctions (e.g. *but*), and discourse adverbials (e.g. *however*), while BioDRB is also annotated for a forth syntactic class – subordinators (e.g. *by*).

There are 100 unique connective types in PDTB (after connectives like *1 year after* are stemmed to *after*) in 18,459 explicit discourse relations (see Table 4.1). Whereas in BioDRB there are 123 unique connective types in 2,636 relations. According to the discourse connective analysis in [55], the subordinators comprise 33% of all connective types in BioDRB. Additionally, 11% of connective types in common syntactic classes that occur in BioDRB do not occur in PDTB; e.g. *In summary*, *as a consequence*. Thus, only 56% of connective types of BioDRB are common to both corpora. While in-domain discourse connective detection has good performance [56], this difference makes the cross-domain identification of discourse connectives a hard task, which is exemplified by experiments in [56] ($F_1 = 0.55$).

With respect to relation sense classification, the connective surface provides already high baselines [54]. However, cross-domain sense classification experiments indicate that there are significant differences in the semantic usage of connectives between two domains, since the performance of the classifier trained on PDTB does not generalize well to BioDRB ($F_1 = 0.57$).

Connective Types	%
Subordinators	33%
Not in PDTB	11%
Common with PDTB	56%
Total	100%

Table 4.2: Differences between PDTB and BioDRB with respect to discourse connective types.

To sum up, the corpora differences with respect to discourse connective usage affect the cross-domain generalization of connective detection and sense classification tasks negatively. The experiments in this chapter are intended to evaluate the generalization of argument span extraction, assuming that the connective is already identified.

4.3 Experiments and Results

The experimental settings for PDTB remain the same and are the following: Sections 02-22 are used for training and Sections 23-24 for testing. For BioDRB, on the other hand, 12 fold cross-validation is used (2 documents in each fold, since in BioDRB there are 24 documents).

Since, unlike PTB for PDTB, for BioDRB there is no manual sentence splitting, tokenization, and syntactic tree annotation; the precise cross-domain evaluation of *Argument Position Classification* and *Argument Span Extraction* steps is not possible. However, we estimate the performance using automatic sentence splitting.

The evaluation methodology remains the same as in Chapter 3. However, we do not propagate error in cross-domain evaluation on BioDRB, since there is no reference information. Additionally, while *Arg1* span extraction models are trained on Gold *Arg2* features, for testing they are always automatic.

4.3.1 Cross-Domain Argument Position Classification

As it was mentioned above, there is no manual sentence splitting for BioDRB; thus, there is no references for whether a discourse relation has its *Arg1* in the same or different sentences. In order to evaluate cross-domain

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
Gold						
SS	90.36	87.49	88.90	70.27	66.67	68.42
PS	79.01	77.10	78.04	46.23	36.61	40.86
ALL	85.93	83.45	84.67	61.94	54.98	58.25
Auto						
SS	86.83	85.14	85.98	64.26	63.01	63.63
PS	75.00	73.67	74.33	37.66	37.00	37.33
ALL	82.24	80.69	81.46	53.93	52.92	53.42

Table 4.3: In-domain performance of the PDTB-trained argument span extraction models on the test set with ‘Gold’ and ’Automatic’ sentence splitting, tokenization, and syntactic features. The results are reported together with the error propagation from argument position classification for Same Sentence (SS), Previous Sentence (PS) models and joined results (ALL) as precision (P), recall (R) and F-measure (F1).

argument position classification we evaluate classifier decisions against automatic sentence splitting using Stanford Parser [34] on whole of BioDRB.

The BoosTexter model for Argument Position Classification, described in Chapter 3 has a high in-domain performance of 97.81. On BioDRB its performance is 95.26, which is still high. Thus, we can conclude that argument position classification generalizes well cross-domain, and that it is little affected by the presence of ‘subordinators’ that were not annotated in PDTB.

4.3.2 In-Domain Argument Span Extraction: PDTB

The in-domain (PDTB) performance of the argument span extraction models, trained on sections 02-22 and tested on sections 23-24 is given on Table 4.3. The results are for 2 settings: ‘Gold’ and ‘Auto’. In the ‘Gold’ settings the sentence splitting, tokenization and syntactic features are extracted from PTB, and in the ‘Auto’ they are extracted from automatic parse trees obtained using Stanford Parser [34].

The general trend in the literature, is that the argument span extraction for *Arg1* has lower performance than for *Arg2*, which is expected since *Arg2* position is signaled by a discourse connective. Additionally, Previous Sentence *Arg1* model performance is much lower than that of the other models

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
SS	80.94	79.88	80.41	66.51	61.82	64.07
PS	82.99	82.99	82.99	57.50	55.62	56.53
ALL	81.45	80.67	81.06	63.87	60.00	61.87

Table 4.4: In-domain performance of the BioDRB-trained argument span extraction models. Both training and testing are on automatic sentence splitting, tokenization, and syntactic features. The results are reported for Same Sentence (SS) and Previous Sentence (PS) models, and the joined results for each of the arguments (ALL) as average precision (P), recall (R), and F-measure (F1) of 12-fold cross-validation.

due to the fact that it only considers immediately previous sentence; which, as was mentioned earlier, covers only 71.7% of the inter-sentential relations. In the next subsections, these models are evaluated on biomedical domain.

4.3.3 In-Domain Argument Span Extraction: BioDRB

In order to evaluate PDTB-BioDRB cross-domain performance we first evaluate the in-domain BioDRB argument span extraction. Since there is no gold sentence splitting, tokenization and syntactic trees, the models are trained using the features extracted from automatic parse trees. We use exactly the same feature sets as for PDTB models, which are optimized for PDTB. An important aspect is that in BioDRB the connective senses are different: there are 16 top level senses that are mapped to 4 top level PDTB senses. For the in-domain BioDRB models, all 16 senses are used.

Since we do not have gold argument position information we do not train in-domain argument classification model. Thus, the reported results are without error propagation. Later, this will allow us to assess cross-domain argument span extraction performance better.

The results reported in Table 4.4 are average precision, recall and f-measure of 12-fold cross-validation. With respect to automatic sentence splitting, there are 717 inter-sentential and 1,919 intra-sentential relations (27% to 73%). Thus, BioDRB is less affected by PS *Arg1* performance than PDTB models, where the ratio is 619 to 976 (39% to 61%). Additionally, BioDRB PS *Arg1* performance is generally higher than that of PDTB. Overall, in-domain BioDRB argument extraction model performance is in-

	Arg2			Arg1		
	P	**R**	**F1**	**P**	**R**	**F1**
Gold						
SS	80.37	76.58	78.42	60.82	56.40	58.52
PS	80.73	80.50	80.62	57.74	52.95	55.19
ALL	80.53	77.71	79.09	59.76	55.29	57.43
Auto						
SS	77.60	75.05	76.30	60.76	55.21	57.83
PS	81.39	81.23	81.31	57.71	51.72	54.47
ALL	78.72	76.80	77.74	59.60	54.12	56.71

Table 4.5: Cross-domain performance of the PDTB-trained argument span extraction models on BioDRB. For the 'Gold' setting the models from in-domain PDTB section are used. For 'Auto', the models are trained on automatic sentence splitting, tokenization, and syntactic features. The results are reported for Same Sentence (SS) and Previous Sentence (PS) models, and the joined results for each of the arguments (ALL) as average precision (P), recall (R), and F-measure (F1) of 12-fold cross-validation.

line with the PDTB models, with the exception that previous sentence *Arg2* has higher performance than the same sentence one.

4.3.4 Cross-Domain Argument Span Extraction: PDTB - BioDRB

Similar to in-domain BioDRB argument span extraction, we perform 12 fold cross-validation for PDTB-BioDRB cross-domain argument span extraction. The cross-domain performance of the models described in previous sections is given in the Table 4.5 under the 'Gold'. To make the cross-domain evaluation settings closer to the BioDRB in-domain evaluation, we additionally train PDTB models on the automatic features. Similar to the in-domain BioDRB evaluation, results are reported without error propagation.

The first observation from cross-domain evaluation is that argument span extraction generalizes to biomedical domain much better that the discourse parsing subtasks of discourse connective detection and relation sense classification. Unlike those subtasks, the difference between in-domain BioDRB argument span extraction models and the models trained on PDTB is much less. The difference between the models trained on automatic and

	Arg2			Arg1		
	P	**R**	**F1**	**P**	**R**	**F1**
Baseline						
SS	80.37	76.58	78.42	60.82	56.40	58.52
PS	80.73	80.50	80.62	57.74	52.95	55.19
ALL	80.53	77.71	79.09	59.76	55.29	57.43
Syntactic						
SS	82.00	75.03	78.33	61.07	51.80	56.01
PS	75.56	74.47	75.01	56.64	46.66	51.11
ALL	80.31	74.98	77.54	59.69	50.42	54.63
No Relation Sense						
SS	81.35	74.00	77.47	62.46	56.11	59.10
PS	80.35	80.13	80.24	57.58	52.25	54.74
ALL	81.16	75.67	78.30	60.86	54.87	57.69

Table 4.6: Cross-domain performance of the PDTB-trained argument span extraction models on BioDRB. For the 'Syntactic' setting the models are trained on only syntactic features (POS-tag + IOB-chain) and 'connective labels'. For 'No Relation Sense', the models are trained by replacing connective sense with 'connective labels'. The 'Baseline' is repeated from Table 4.5. The results are reported for Same Sentence (SS) and Previous Sentence (PS) models, and the joined results for each of the arguments (ALL) as average precision (P), recall (R), and F-measure (F1) of 12-fold cross-validation.

gold parse trees is also not high, and gold feature trained models perform better with the exception of PS *Arg2*. Since training on automatic parse trees does not improve cross-domain performance, the rest of the experiments is using gold features for training.

4.3.5 Feature-Level Domain Adaptation

The two major differences between PDTB and BioDRB are vocabulary and connective senses. The out-of-vocabulary rate of PDTB on the whole BioDRB is 22.7% and of BioDRB on PDTB is 33.1%, which are very high. Thus, PDTB lexical features might not be very effective, and the models generalize well due to syntactic features. To test this hypothesis we train additional PDTB models on only syntactic features: POS-tags and IOB-chain and 'connective labels' – 'CONN' suffixed for the Beginning (B), Inside (I) or End (E) of the connective span, simulating discourse connective detection output. Moreover, we reduce the feature set to **unigrams** only (recall that features were enriched by 2 and 3 grams), such that the models become very general.

Even though BioDRB connective senses can be mapped to PDTB, in the published works it was observed that relation sense classification does not generalize well. To reduce the dependency of argument span extraction models on relation sense classification, the connective sense feature is also replaced by 'connective labels'. We train these models using gold features only, and, similar to previous experiments, do 12-fold cross-validation.

The performance of these adapted models is given in Table 4.6. The 'Syntactic' section gives the results of the models trained on syntactic features and the 'No Relation Sense' section gives the results for the models with 'connective labels' instead of connective senses, and the 'Baseline' repeats the performance of the PDTB-optimized models described in previous section.

The PDTB-optimized baseline outperforms the domain adapted (generalized) models on *Arg2*; however, 'No Relation Sense' *Arg1* yields the best performance, and, though insignificantly, outperforms the baseline. Thus, the effect of replacing connective senses with 'connective labels' is negative for all cases except SS *Arg1*. Overall, the difference in performance between the Baseline and 'No Relation Sense' models is an acceptable price to pay for the independence from relation sense classification.

The most general models – unigrams of Part-of-Speech tags and IOB-chains together with 'connective labels' in the window of ±2 tokens – all have the performance lower than the baseline, which is expected given its feature set. However, for the easiest case of intra-sentential *Arg2* it outperforms the model trained by replacing the connective sense in the baseline (i.e. 'No Relation Sense'). Degraded performance of *Arg1* models indicates that lexical features are helpful.

Introducing the tokens back into the model, and increasing the features to include also 2-grams, boosts the performance of the models to outperform the 'No Relation Sense' models in all but Previous Sentence *Arg2* category. However, the models now yield performance comparable to the PDTB optimized baseline (insignificantly better), while being unaffected by poor cross-domain generalization of relation sense classification (Table 4.7).

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
SS	81.72	76.14	78.82	61.53	56.36	58.82
PS	80.31	79.84	80.07	58.55	52.82	55.44
ALL	81.27	77.10	79.12	60.56	55.30	57.80

Table 4.7: Cross-domain performance of the PDTB-trained argument span extraction model on unigram and bigrams of token, POS-tag, IOB-chain and 'connective label'. The results are reported for Same Sentence (SS) and Previous Sentence (PS) models, and the joined results for each of the arguments (ALL) as average precision (P), recall (R), and F-measure (F1) of 12-fold cross-validation.

The cross-domain argument extraction experiments indicate that models trained on PDTB-optimized feature set already have good generalization. However, they are dependent on relation sense classification task, which does not generalize well. By replacing connective senses with 'connective labels' we obtain models independent of this task while maintaining comparable performance. The in-domain trained BioDRB models, however, perform better, as expected.

4.4 Conclusion

In this chapter we presented cross-domain discourse parser evaluation on subtasks of argument position classification and argument span extraction. The observed cross-domain performances are indicative of good model generalization. However, since these models are applied later in the pipeline, they are affected by the cross-domain performance of the other tasks. Specifically, discourse connective detection, which in literature was shown not to generalize well. Additionally, we have presented feature-level domain adaptation techniques to reduce the dependence of the cross-domain argument span extraction on other discourse parsing subtasks.

The syntactic parser (Stanford) that also provides sentence splitting and tokenization is trained on Penn Treebank, i.e. it is in-domain for PDTB and out-of-domain for BioDRB. Also it is known that domain-optimized tokenization improves performance on various NLP tasks. Thus, the future direction of this work is evaluate argument span extraction using tools optimized for biomedical domain.

Chapter 5

Cross-Language Porting of Spoken Language Understanding[1]

Automatic cross-language Spoken Language Understanding (SLU) porting is plagued by two limitations. First, SLU are usually trained on limited domain corpora. Second, language pair resources (e.g. aligned corpora) are scarce or unmatched in style (e.g. news vs. conversation). In this chapter we present experiments on automatic SLU porting using SMT and apply existing approaches to the problem – Test-on-Source and Test-on-Target – to LUNA Corpus. We evaluate SLU porting on close and distant language pairs: Spanish - Italian and Turkish - Italian; and in-domain and out-of-domain SMT systems.

5.1 Introduction

With respect to the direction and the object of translation, the approaches to Spoken Language Understanding (SLU) porting via Statistical Machine Translation (SMT) can be grouped under two categories: Test-on-Source and Test-on-Target. In the *Test-on-Source* approach the direction of translation is from a language the system is being ported to (target language) to

[1] The Chapter is partially based on:

1. E.A. Stepanov, I. Kashkarev, A.O. Bayer, G. Riccardi, and A. Ghosh. "Language Style and Domain Adaptation for Cross-Language SLU Porting", *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2013. [64]
2. E.A. Stepanov, G. Riccardi and A.O. Bayer. "The Development of the Multilingual LUNA Corpus for Spoken Language System Porting", LREC, Reykjavik, 2014 [67].

the language of the existing SDS (source language). The object of translation is user utterances in the target language. Consequently, SLU of the existing system is "extended" via SMT to cover a new language, and the success depends on the quality of machine translation. In the *Test-on-Target* approach (also referred to as Train-on-Target) the direction of translation is the opposite, i.e. from the source language to the target language. The object of translation is the data used to train the source SLU, and new language understanding components are trained. Thus, the success also relies on the accurate transfer of annotation.

In the literature, the Test-on-Source approach is credited as having better performance (e.g. [29, 30, 31, 40]). Moreover, the procedure is simpler to implement, since it does not require porting of annotation. Additional techniques such as statistical post-editing and 'smeared' SLU training proposed by [31] make this approach even more appealing.

Both approaches to SLU porting suffer from two limitations: (1) SLU are usually trained on limited domain corpora, and (2) parallel corpora are scarce. Majority of the research on SLU porting make use of in-domain parallel corpora (usually manually translated) to train SMT systems, and experiment on close or resource-rich language pairs. This chapter, on the other hand, presents Test-on-Source and Test-on-Target SLU porting via SMT using off-the-shelf general-domain system and a system trained on out-of-domain data. We evaluate end-to-end SLU system porting on both close and distant language pairs: Spanish - Italian and Turkish - Italian. There systems are developed further in Chapter 7, here, on the other hand, we first describe corpora and SMT systems used throughout the thesis and provide the baseline results.

5.2 Corpora

In this section we briefly describe the corpora used to train SMT systems.

The **in-domain** corpus used throughout the Chapters 5 to 7 is *Multilingual LUNA Corpus* [67], which is the translation of *Italian LUNA Corpus* [15] to Spanish, Turkish, and Greek (see Chapter 2). The Multilingual LUNA Corpus is used to train in-domain SMT system.

The **out-of-domain** *Europarl Parallel Corpus* [36] of the proceedings of the European Parliament is the most popular corpus in machine translation community. It encompasses 21 European languages, including the languages of interest: Spanish and Italian. Version 7 (from May 2012) of the corpus was used to create Italian-Spanish parallel corpus of approximately 1.8M sentence pairs. This parallel corpus is used to train the out-of-domain Spanish-Italian SMT system.

5.3 Baseline SMT Systems

Google Translate is a **general-domain** SMT system designed to provide reliable translations of text in multiple genres. It is trained on a vast variety of parallel written texts (as opposed to speech transcriptions). Since it is targeted for a wide range of languages, the translations go through English as a bridge language, i.e. a sentence in Turkish or Spanish is first translated into English and then to Italian.

Europarl Moses is an **out-of-domain** data trained moses-based SMT system. Moses[2] is a statistical machine translation system that, given a parallel corpus, allows training translation models for any language pair automatically. The tool supports various translation models: phrase-based and tree-based, as well as factored models; and input of different level of complexity from text to ASR lattices. Here we use a phrase-based translation model on plain text. Prior to the training, Europarl corpus was pre-processed to be suitable for speech transcriptions: it was tokenized, lowercased and all punctuation was removed.

LUNA Moses is an **in-domain** data trained SMT system. Multilingual LUNA Corpus was used to train both Spanish - Italian and Turkish - Italian systems. These systems represent an upper-bound performance.

The most widely used evaluation metric for the statistical machine translation is the BLEU score (Bilingual Evaluation Understudy) [49]. It is known to correlate well with human judgments on the translation quality and as it turns out with further SLU evaluation on the translated sentences. The BLEU score is the weighted sum of precision of n-grams.

[2] http://www.statmt.org

SMT System	*Language Pair*	
	ES-IT	**TR-IT**
Google Translate	25.89	13.72
Europarl Moses	35.08	N/A
LUNA Moses	49.77	33.39

Table 5.1: SMT System Baselines: *Google Translate*: General-domain Off-the-Shelf SMT; *Europarl Moses*: Out-of-domain SMT; *LUNA Moses*: In-domain SMT. Performance on LUNA Development Set for Spanish-Italian (ES-IT) and Turkish-Italian (TR-IT) is reported as 4-gram BLEU score.

The metric is designed to evaluate both adequacy (how much information is transferred between the original and the translation) and fluency (how good the target language output is) of translation. Adequacy is captured by shorter n-gram matches (1-2-grams), whereas fluency is captured by longer n-grams matches. (3 and higher n-grams). All the results reported are of 4-gram BLEU score.

The performance of the three baseline SMT systems for Spanish - Italian and Turkish - Italian language pairs is reported in Table 5.1 using 4-gram BLEU score [49]. Since Europarl does not have Turkish, there is no out-of-domain STM system for Turkish - Italian. As expected, in-domain SMT systems perform the best for both language pairs, followed by the out-of-domain SMT system for Spanish - Italian, since the training corpus was already pre-processed for speech transcriptions. Google Translate has the worst performance.

5.4 Spoken Language Understanding Module

In this work we do not develop a Spoken Language Understanding Model, rather we utilize the baseline model of [3]. Similar to the Discourse Parser the SLU models are based on conditional random fields (CRF) [38]. CRFs are discriminative undirected graphical models which have been successfully used for segmenting and labeling sequential data. CRFs model the conditional probability of the concept sequence given the word sequence.

The Italian LUNA Spoken Language Understanding model of [3], as well as SLU models for other languages that are used throughout the experiments are trained using the following types of features:

- *Orthographic*: first and last n letters of a token, where n ranges from 1 to 5 (10 features);
- *Ngrams*: unigrams and bigrams of tokens in the window of ± 1 tokens, including $-1, 1$ token pair (6 features);
- *Binary*: a feature to label numerical expressions (1 feature);

All the features are independent in the window of ± 1 tokens. Additionally, CRFs use the previous output token as a feature for current token decision.

5.4.1 SLU Evaluation

A commonly accepted metric for SLU evaluation is Concept Error Rate (CER), which is based on the Levenshtein alignment of sentences and computed as the ratio between inserted, deleted and substituted concepts and the total number of concepts in the reference sentence. The SLU model trained on original LUNA Corpus has Concept Error Rate (CER) of 21.5%.

5.5 Test-on-Source SLU

In the Test-on-Source approach there is already an SLU model in the source language and SMT is deployed to translate the target language utterances to the source language. For the two target languages, Spanish and Turkish, utterances are translated to Italian, using the SMT systems described above. The translated utterances are the input to the SLU for semantic parsing (extraction of domain concepts).

Table 5.2 reports the SLU performance of the baseline SMT systems in terms of CER. The results indicate that in the Test-on-Source scenario the language distance is an important factor; since, Spanish SMTs yield much lower CER. Another expected observation is that the in-domain data trained SMT results in better SLU performance irrespective of language distance.

SMT System	**CER**
Spanish - Italian	
Google Translate	43.00
Europarl Moses	39.20
LUNA Moses	25.80
Turkish - Italian	
Google Translate	56.90
LUNA Moses	39.20

Table 5.2: The Test-on-Source SLU performance of the baseline SMT systems on the LUNA Test Set.

5.6 Test-on-Target SLU

Unlike the Test-on-Source approach, in Test-on-Target approach SMT is deployed to translate the source language corpora to the target language. For the two target languages, Spanish and Turkish, SMT is used to transfer annotation from Italian, and the target language SLU systems are trained. The annotation transfer is performed using word alignment following the approaches described in [31].

- **Direct Alignment**: The alignment is generated by mapping source language concepts to the target language utterance *directly*, i.e. no source language utterance is involved.

- **Indirect Alignment**: The concept tags in the source language are projected via word-to-word alignment generated using utterances.

- **Alignment via Tagged Translation**: The source language utterances are automatically translated to the target language constraining the word reordering, i.e. annotated concepts are not broken. This is achieved by inserting XML tags.

Table 5.3 presents the results of Test-on-Target on LUNA Development and Test Sets for the three alignment approaches for Italian - Spanish and Italian - Turkish. In line with the results of [31], the indirect alignment performs the best.

Alignment	**DEV**	**TEST**
Italian-Spanish		
Direct	52.2	45.1
Indirect	31.6	29.0
Tagged	35.9	31.7
Italian-Turkish		
Direct	64.0	58.7
Indirect	49.9	46.5
Tagged	52.9	47.6

Table 5.3: Test-on-Target SLU performance of the SMT systems trained on Multilingual LUNA Corpus using different alignment models. Performance is reported on the LUNA Development and Test Sets as Concept Error Rate (CER).

5.6.1 Relaxing the References

Due to the fact that Multilingual LUNA Corpus lacks manual concept annotation in the target languages, the results presented in Table 5.3 are the obtained via evaluation against Italian references. However, since word order of distant languages, such as Turkish, affects the order of concepts, this evaluation is very restrictive. To overcome this restriction, we additionally evaluate Test-on-Target SLU performance obtained through the best annotation transfer approach – Indirect Alignment – on the following settings:

- *Sorted References*: Both hypothesis and reference are sorted alphabetically.

- *Transferred References*: The reference set in this case is obtained through annotation transfer, similar to the training set.

- *Transferred and Sorted References*: The references in the previous setting are additionally sorted alphabetically.

Table 5.4 presents the results on these evaluation settings. The fact that the performance does not change much for Spanish SLU, but it varies greatly for Turkish SLU, confirms that, language distance, i.e. the word order differences, has a significant impact on SLU Porting.

References	**Spanish**	**Turkish**
Italian	31.6	49.6
Sorted	31.6	41.8
Transferred	31.8	43.5
Transferred and Sorted	31.6	40.5

Table 5.4: Test-on-Target SLU performance of the SMT systems trained on Multilingual LUNA Corpus using indirect alignment models, evaluated on different references. Performance is reported on the LUNA Development Set as Concept Error Rate (CER).

Phrase Table	**DEV**	**TEST**
BLEU		
LUNA	55.46	57.76
LUNA Reduced	52.90	54.03
Europarl	36.57	34.40
Europarl Reduced	35.33	31.50
CER		
LUNA	6.0	4.7
LUNA Reduced	5.3	4.3
Europarl	5.7	4.1
Europarl Reduced	4.9	3.1

Table 5.5: Reduced Translation Model evaluation on in-domain and out-of-domain data training. Performance is reported on the LUNA Development and Test Sets as BLEU and Concept Error Rate (CER).

5.6.2 Out-of-Domain Corpus Annotation Transfer

In-domain spoken language corpora might not be available; thus, it makes sense to evaluate the annotation transfer approaches on the alignment models produced on out-of-domain data. Since Turkish is not in Europarl, this set of experiments is performed for Italian - Spanish Test-on-Target SLU. The performances are evaluated using Italian references.

Due to the fact that in Spoken Language Understanding concepts are relatively short segments of text, it is possible to constrain the SMT translation table to those segments only; additionally, leaving word-to-word pairs. The *reduced* phrase table is expected to prevent unnecessary reordering. Four settings are evaluated: in-domain LUNA Corpus: full and reduced translation tables, and Europarl: full and reduced translation tables. The performances of translation models are given in Table 5.5. The

Phrase Table	**CER**
LUNA	36.1
LUNA Reduced	34.8
Europarl	40.8
Europarl Reduced	41.0

Table 5.6: Reduced Translation Model Test-on-Target SLU evaluation. Performance is reported on the LUNA Development Set as Concept Error Rate (CER).

evaluation is both in terms of BLEU score and Concept Error Rate. Concept Error Rate evaluation uses Italian references.

Reducing the phrase table to domain concept affects the BLEU score negatively; but the cross-language annotation transfer improves. The effect is similar for both in-domain and out-of-domain models. However, while the annotation transfer via reduced in-domain phrase table yields better Test-on-Target performance (see Table 5.6), the performance of the SLU trained on data produced via annotation transfer using reduced out-of-domain phrase table is affected negatively.

The Test-on-Target experiments presented in this Section indicate that the annotation projection methodology is useful for the annotation of the target language data; since the transfer error is relatively low. However, training the target language models on the automatically created data, in context of Spoken Language Understanding is inferior to the Test-on-Source approach.

5.7 Conclusion

In this chapter we have presented our approach to SLU and SMT Systems. We have evaluated Test-on-Source and Test-on-Target approaches to SLU porting on LUNA Corpus. We have demonstrated that, in line with published works, Test-on-Source approach yields better performance. The approach is further developed in Chapter 7. It is expended with Language-Style and Domain Adaptation.

Chapter 6

Cross-Language Transfer of Semantic Annotation via Targeted Crowdsourcing[1]

An alternative to the automatic cross-language transfer of semantic annotation via Statistical Machine Translation (SMT) is the transfer of annotation via crowdsourcing. However, the current crowdsourcing approach faces several problems. First, the available crowdsourcing platforms have skewed distribution of language speakers. Second, speech applications require domain-specific knowledge. Third, the lack of reference target language annotation, makes crowdsourcing worker control very difficult. In this chapter we address these issues on the task of cross-language transfer of domain-specific semantic annotation from an Italian spoken language corpus to Greek, via *targeted* crowdsourcing. The issue of domain knowledge transfer is addressed by priming the workers with the source language concepts. The lack of reference annotation is coped with a consensus-based annotation algorithm. The quality of annotation transfer is assessed using source language references and inter-annotator agreement. We demonstrate that the proposed computational methodology is viable and achieves acceptable annotation quality.

[1] The Chapter is published as S.A. Chowdhury, A. Ghosh, E.A. Stepanov, A.O. Bayer, G. Riccardi, and I. Klasinas. "Cross-Language Transfer of Semantic Annotation via Targeted Crowdsourcing", INTERSPEECH, 2014 [12].

6.1 Introduction

An important step in the development of a spoken language understanding system is semantic annotation of speech utterance transcriptions. A brute force approach to multilingual porting of speech applications would require the replication of this process for each target language. While the text of a corpus can be translated from the source language to the languages of interest using translation services, transfer of its annotation remains a research issue. Crowdsourcing – a recent computational model for large-scale distributed task execution – has the potential to be the solution. However, the feasibility of the semantic annotation via crowdsourcing is affected by factors such as the language of interest, the domain-specificity of the required annotation, and the availability of the resources for the evaluation of the crowd-annotated data.

First, the language of interest might be under-represented on existing crowdsourcing platforms due to the skewed worker demographics. Consequently, obtaining sufficient amount of adequately annotated data is an issue. An alternative is to access language speaker groups via other channels, and to design tasks *targeted* to that specific language groups.

Second, the semantic annotation required for speech applications is usually domain-specific. For example, for Information Technology domain a worker might be required to distinguish between hardware, software and network operations. Due to the fact that there is none to a minimal amount of time to provide some domain knowledge to the workers, the level of domain-specificity of the required annotation increases the complexity of the task, and it is expected to decrease the quality. Thus, the domain knowledge has to be transferred by other means. Researchers have successfully used live-feedback signals to improve the performance of the workers in crowdsourcing [17, 57]. In this work, on the other hand, the workers are *primed* with the source language concepts.

Third, the traditional method for quality control in crowdsourcing tasks require the annotations to exist in a target language, which is not always the case. Coupled with the constraints imposed by the limited number of workers for low-resource languages, the traditional evaluation methodology is not applicable. We approach the problem of evaluation of annotation using inter-annotator agreement and the source language references. Traditional metrics for inter-annotator agreement are designed for a fixed number of annotator over a fixed data-set; thus, the evaluation of the quality of crowdsourced annotation without expert references is still an open question. Additionally, in cross-lingual tasks language distance is an important factor: since source language references may be reused for close languages and not for distant ones due to the word order and concept representation differences.

These issues are addressed on the task of cross-language transfer of domain-specific semantic annotation from Italian to Greek in spoken language corpus via *targeted* crowdsourcing. The language pair represents distant languages, which are under-represented on popular crowdsourcing platforms. The semantic annotation task requires workers to make two decisions – on the span of the concept and its label; thus, there is a task of *concept segmentation* as well as cross-language transfer of domain-specific *concept labels.*

The chapter is structured as follows. In Section 6.2 we briefly review related works on cross-language annotation transfer and crowdsourced annotation. In Sections 6.3 and 6.4 we describe the DIY *targeted* crowdsourcing platform and the crowdsourced cross-language annotation transfer task design, respectively. Section 6.5 presents the evaluation methodology and the results. Section 6.6 provides concluding remarks.

6.2 Related Work

The cross-language annotation transfer in the literature was successfully applied to a variety of tasks via Statistical Machine Translation (SMT) methods [78, 59, 4, 48]. In the context of semantic annotation for spoken language application, the SMT methodology was applied in [31] to transfer semantic annotation from French to Italian. The general idea of the approach is presented in Figure 6.1 that depicts Italian-Greek phrase alignment and the annotation transfer.

The annotation transfer via SMT requires parallel corpora, and its evaluation requires expert annotated resources. However, it is costly to obtain

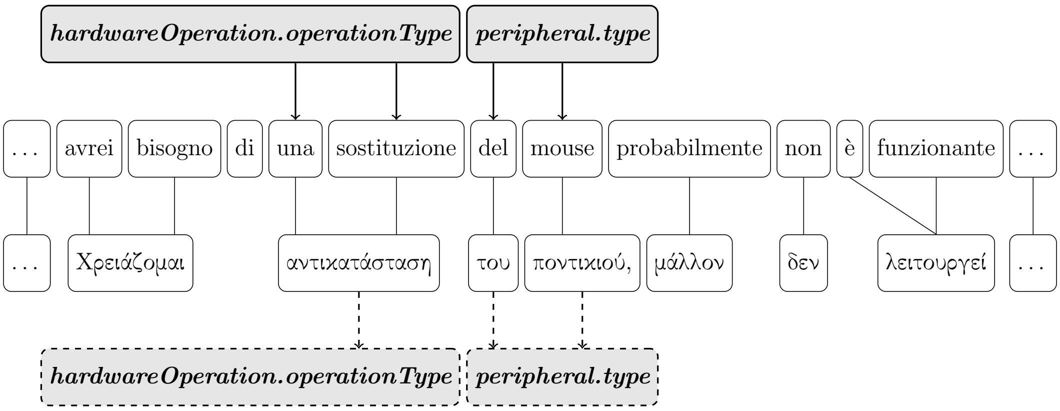


Figure 6.1: General idea of cross-language annotation transfer. Italian and Greek utterances are not one-to-one aligned. A concept can be linked to a single word in Greek, but multiple words in Italian or vice versa.

expert annotation for each language. To overcome this, we use crowdsourcing for cross-language transfer of semantic annotation and apply inter-annotator agreement as a measure of within target language annotation quality; and evaluate the annotations against source language references as a measure of cross-language transfer quality.

In recent years crowdsourcing has been successfully applied to a variety of research problems. The mechanism is usually ideal for performing tasks that can be broken into microtasks and distributed to a crowd of workers. In the Natural Language Processing (NLP) domain it has been used for corpus creation [9, 47] , transcription [43, 50], translation [79], and annotation tasks [20, 27]. On the other hand, we apply crowdsourcing for cross-language annotation transfer, which is different from general annotation, because the workers are provided with a set of concepts that exist in the utterance in the source language.

6.3 Targeted Crowdsourcing

The main challenge of generalistic human computation platforms such as Amazon Mechanical Turk is attracting a large number of qualified workers to participate in tasks while filtering out low quality workers and spammers. Since enrollment to such platforms does not require any particular skill set from workers, it is up to the task designers to overcome this issue.

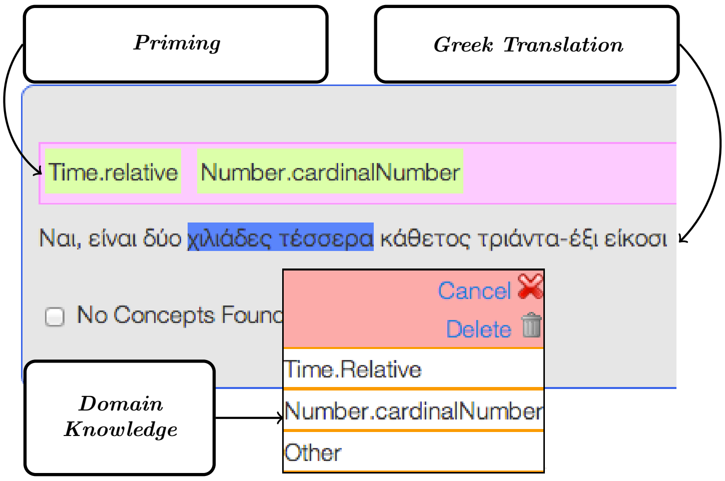
Traditionally, in research community this problem is solved using qualification tests, gold standard evaluation on selected items of the task [50], and other techniques to penalize low quality work. Additionally, the pseudo-anonymity of the workers enforced by most crowdsourcing platforms makes it difficult to target workers or worker-groups with the desired skill set.

Targeted crowdsourcing has evolved as a new paradigm with the intent to overcome this drawback. In targeted crowdsourcing the objective is to attract workers who are likely to have the skills needed for the target task and to design the platform appropriately. Crowdsourcing for creative ideas and problem solving are firm examples. For example, in enterprise settings, a crowd of employees was successfully used to improve the overall business process of the company [72]. Recently, the US Centers for Disease Control and Prevention (CDC) launched the CDCOLOGY project [10], a microvolunteering platform, targeting the population of registered university students. As an example of targeting a more special skill set, Open Mind Word Expert [11], a volunteer-based web framework to tag words with appropriate senses from WordNet, has been able to attract enough volunteers with sufficient proficiency for the tasks.

For the task of semantic annotation transfer from one language to another, the required skill is the target language proficiency (Greek). The demographic distribution of workers on platforms such as Amazon Mechanical Turk is very skewed: close to 90% of turkers are from US and India [60]. Hence, the utility of the platform is low for NLP tasks involving languages of under-represented speaker groups. In collaboration with researchers from target language speaking institutions a targeted crowdsourcing experiment was carried out.

6.4 Semantic Annotation Transfer Task

The Multilingual LUNA Corpus [67, 64], described in Chapter 2, was used for crowdsourced annotation transfer task. As it was described earlier, the corpus is the translation of Italian LUNA Corpus [15] to Spanish, Turkish and Greek via professional translation services. The translations are plain text, i.e. the semantic annotation have not been transferred.



Ναι, είναι δύο χιλιάδες τέσσερα κάθετος τριάντα-έξι είκοσι

Figure 6.2: Description of each task. For each Greek utterance, the concepts from the source language (Italian) are used for priming. The domain knowledge is transferred using the LUNA concept ontology.

6.4.1 Task Design

A set of 800 Greek utterances from the Multilingual LUNA Corpus [67] was put up for crowdsourcing. Each worker had to annotate 50 utterances presented on 5 pages (10 utterances per page).

The task had concise instructions and a short video demonstrating the annotation process to workers. Since Greek translations lack both segmentation and concept labels; the worker had to perform two subtasks: concept segmentation and labeling. After reading an utterance, a worker had to highlight a segment of an utterance covering a single concept and select the most suitable label from a drop-down menu (See Figure. 6.2).

The LUNA concept ontology contains a total of 45 unique concepts arranged in a two-level hierarchy with 26 top-level concepts. To ease the concept selection, the drop-down menu of concepts was arranged with respect to this 2-level hierarchy. No overlaps or nesting of concepts is allowed. However, a worker could mark an utterance as containing no concepts.

6.4.2 Priming the Workers

The semantic information is mostly preserved during the process of translation [4]. Consequently, the concepts from the Italian references were provided to the workers in the form of a unique list of suggested concepts on top of each utterance. The idea behind priming is to transfer the knowledge of the domain and provide a worker with semantic information to support the annotation task. The workers were free to highlight and mark segments matching the suggested concepts or ignore the list entirely.

6.5 Results and Discussion

In this Section we provide the details about the annotated data collected via crowdsourcing task described in the previous Section; and evaluate the quality of the annotations.

6.5.1 Data Collection Results

Fifty workers completed over 2000 micro-tasks over a period of two weeks. From the subset of 800 annotated utterances, 536 were annotated by at least three workers. The number of annotated concepts between languages differs: while there are 2,227 concepts in the references (Italian), there are on average 1,439 (35% less) concepts in Greek. Comparison between the suggested and the annotated concepts indicates that 44% of suggested concepts were ignored by the workers; while 9% of annotated concepts were not from the suggested lists.

For the evaluation we consider only utterances that have at least three judgments (536 utterances). We first evaluate the inter-annotator agreement between the workers, and then the transfer of annotation between languages.

6.5.2 Inter-Annotator Agreement

We first describe the evaluation methodology and then the agreement on the two subtasks of semantic annotation individually and together.

Evaluation Methodology

The commonly accepted metric for the assessment of the quality of an annotated resource is to measure the agreement between annotators. The most widely used agreement measure is κ (Cohen's for two and Fleiss' for several annotators), which is a chance corrected percent agreement measure. Unfortunately, κ is designed for a setting with a fixed number of annotators over a fixed data set; and this is not the case in crowdsourcing. Additionally, in text markup tasks, such as annotation, the number of *true negatives*, required for the calculation of the observed and chance agreements in κ, is not well defined (e.g. the number of text segments discarded by the workers as concept chunks). These factors make κ impractical as a measure of agreement of crowdsourced annotation.

An alternative agreement measure that does not depend on *true negatives* is Positive (Specific) Agreement [21], which is identical to the widely used F-measure [26]. Even though the measures are also for the fixed number of annotators on a common data set, since they do not rely on *true negatives* and the chance agreement, they are better suitable for the evaluation of crowdsourced annotation. In our crowdsourcing experiment we have collected 3 judgments per utterance; thus, for computing pair-wise F-measures we randomly assign each judgment to one of the three hypothetical annotators. The reported F-measures are averages of pair-wise F-measures among these three hypothetical annotators.

In text markup tasks annotators might select different spans all of which might be considered correct. For instance, for the *hardware* concept the selected span might be *with the printer*, *the printer*, or only *printer*. Thus, we report results for *exact* and *partial* matches [32]. Since in semantic annotation tasks workers are taking two decisions, we evaluate the agreement on these decisions separately as *segmentation* and *labeling* agreements and jointly as *semantic annotation agreement.*

Segmentation Agreement

Segmentation Agreement is the measure of the agreement of the workers on concept spans regardless of the label they give to the selected span. The averages of pair-wise precision, recall and F-measures are reported for exact and partially matched spans in Table 6.1 (upper part). Agreement on partial matches is relatively low: $F_1 = 63.56$, due to the fact that the measure also considers 'missing' concepts, i.e. identified only by one of

Match	**P**	**R**	**F1**
Whole Data			
Exact	39.60	38.58	39.08
Partial	64.24	62.90	63.56
Common Span Subset			
Exact	46.10	47.41	46.74
Partial	69.16	71.07	70.10

Table 6.1: Segmentation Agreement reported as averages of pair-wise precision (P), recall (R) and F-measures (F1) for exact and partial matches on whole data and the subset of common spans.

	P	**R**	**F1**
Exact	48.39	47.15	47.76
Set	67.71	73.37	70.55

Table 6.2: Labeling Agreement reported as averages of pair-wise precision (P), recall (R) and F-measures (F1) for exact match and set (compares lists of unique concepts regardless of the order)

the annotators. The segmentation agreement on the set of spans common to all of the judgments for an utterance is acceptably higher: $F_1 = 70.10$ (Table 6.1, lower part).

Labeling Agreement

Labeling Agreement is the measure of the agreement of the workers on the concept labels, regardless of the agreement on their spans. Unlike Segmentation Agreement there are no partial matches (each concept is represented by a single token). In order to evaluate the labeling agreement independently from segmentation differences[2] we additionally compute the agreement over sets of annotated concepts.

The labeling agreement results are reported in Table 6.2. The average of pair-wise F-measures for the match (*Exact* in Table 6.2) is 47.76. The average of pair-wise F-measures for the set condition is considerably higher – 70.55. The results indicate that there are also differences in the segmentation of the same concepts.

[2] E.g.: a worker might choose to annotate numerical expressions like *one seven* as a single *number* concept or as two.

Match	**P**	**R**	**F1**
Exact	33.77	32.90	33.32
Partial	51.45	50.35	50.89

Table 6.3: Semantic Annotation Agreement – jointly for segmentation and labeling – reported as averages of pair-wise precision (P), recall (R) and F-measures (F1) for exact and partial matches.

Semantic Annotation Agreement

Semantic Annotation Agreement is the measure that considers both segmentation and labeling. It is the most strict of the inter-annotator agreement measures, since annotators have to agree both on the label and on its span. The results are reported in Table 6.3. The average of pair-wise F-measures for partial matches is only 50.89.

Even though, the inter-annotator agreement is relatively low on each of the subtasks of the semantic annotation, none of the workers is an expert. Thus, these results are indicative only of the variability in annotation. Since the task is a transfer of semantic annotation, there are also the expert annotated source language references. In the next Section we exploit these references to evaluate the quality of transfer and acceptability of the collected annotations.

6.5.3 Cross-Language Annotation Transfer

In this Section we evaluate the transfer of the annotation from the source language (Italian) to the target language (Greek). Similar to the previous subsection, we first present the evaluation methodology and then the results.

Evaluation Methodology

Since the order of concepts might be affected by the differences in the word-order between languages, the cross-language evaluation is carried on the sorted lists of concepts per utterance. We compare the annotated concept labels (i.e. spans are not considered) against the labels in the Italian reference preserving the number of concepts in each case. This evaluation allows us to assess the amount of actual transfer. For the evaluation we

	P	**R**	**F1**
Random Re-sampling	84.40	54.54	66.26
ROVER	83.87	69.82	76.20

Table 6.4: Cross-Language Transfer using random re-sampling and ROVER as precision (P), recall (R) and F-measure (F1); for random re-sampling the results are averages of 1,000 iterations.

randomly select one of the judgments and compute precision, recall, and F-measure using Italian references. The procedure is repeated 1,000 times and the results are averaged.

Recognition Output Voting Error Reduction (ROVER) is one of the most frequently used tool in Automatic Speech Recognition community. The tool combines hypothesized sequence outputs of multiple recognition systems (in this case: workers) and selects the best scoring sequence. We applied the technique to the collected non-expert annotations to produce a single one. Since the three judgments are over the same utterance, we have applied majority voting on token level to decide on the span and the label of concepts (out-of-span tokens are taken as having 'null' label). As a result we obtain a single majority voted annotation hypothesis. Similar to random re-sampling, the output of ROVER is evaluated against Italian references. The expectation is that ROVER improves the overall annotation transfer.

Quality of Transfer

The results for the two evaluation settings – random re-sampling and ROVER – are reported in Table 6.4. The results indicate that even with the inter-annotator agreement of $F_1 = 50.98$ for joint span and label decisions, using techniques such as ROVER, it is possible to exploit 'the power of the crowd' to transfer annotation with acceptable quality. By combining non-expert annotator decisions we gain approximately 15% in recall. Even though, the recall for transferred annotation using ROVER is ≈ 70, the precision is acceptably high ≈ 84.

Overall, the combination of crowdsourcing and computational techniques such as ROVER make the approach viable for the cross-language annotation transfer.

6.6 Conclusion

In this chapter we have addressed the problem of transferring the semantic annotation from the source language corpus (Italian) to a low-resource distant target language (Greek) via crowdsourcing. We have addressed the issue of the skewed language speaker distribution of current crowdsourcing platforms by using targeted crowdsourcing. We have presented the approach to transfer domain knowledge, required for the semantic annotation, via priming with a list of source language concepts. Additionally, we have presented the methodology to assess quality of the crowd annotated corpora using inter-annotator agreement and evaluation against source language references. We have demonstrated that by combining the 'power of the crowd' in the form of multiple hypotheses with a computational method such as ROVER the resulting corpus achieves acceptable annotation quality.

Chapter 7

Language-Style and Domain Adaptation for Cross-Language Porting[1]

Automatic cross-language Spoken Language Understanding (SLU) porting is plagued by two limitations. First, SLU are usually trained on limited domain corpora. Second, language pair resources (e.g. aligned corpora) are scarce or unmatched in style (e.g. news vs. conversation). We present experiments on automatic style adaptation of the input for the translation systems and their output for SLU. We approach the problem of scarce aligned data by adapting the available parallel data to the target domain using limited in-domain and larger web crawled close-to-domain corpora. SLU performance is optimized by re-ranking its output with Recurrent Neural Network-based joint language model. We evaluate end-to-end SLU porting on close and distant language pairs: Spanish - Italian and Turkish - Italian; and achieve significant improvements both in translation quality and SLU performance.

The experiments presented in this chapter are on SLU porting; however, the range of applications of the techniques is much broader. Language Style Adaptation is beneficial for any task involving spoken language corpora and written language tools. The discussed Statistical Machine Translation domain adaptation techniques are novel in their application to SLU port-

[1] The Chapter is published in E.A. Stepanov, I. Kashkarev, A.O. Bayer, G. Riccardi, and A. Ghosh. "Language Style and Domain Adaptation for Cross-Language SLU Porting", *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2013. [64].

ing. The difference of SLU from Discourse Parsing within the context of this chapter is minimal, and only applies to the re-ranking with Recurrent Neural Networks.

7.1 Introduction

As it was already described in Chapter 5, with respect to the direction and the object of translation, the approaches to Spoken Language Understanding (SLU) porting via Statistical Machine Translation (SMT) can be grouped under two categories: Test-on-Source and Test-on-Target. In line with the literature, our experiments in the previous chapter show that the Test-on-Source approach has better performance. Thus, we use this scenario to approach the limitations of cross-language Spoken Language Understanding porting: (1) domain specificity of SLU corpora, and (2) scarcity of parallel corpora.

We present experiments on language style adaptation for off-the-shelf SMT systems and domain adaptation for the SMT systems trained on out-of-domain data. The corpora used for domain adaptation are *in-domain* corpus used to train the source language SLU, and *close-to-domain* web crawled corpus. Both language style and domain adaptation take place in the SMT pipeline. The semantic parses of the translation hypotheses are further re-ranked with in-domain Recurrent Neural Network-based joint language model [2] in the source language (see Figure 7.1 for the overall architecture of the process). The end-to-end Spoken Language Understanding system porting is evaluated on both close and distant language pairs: Spanish - Italian and Turkish - Italian; and the significant improvements are achieved both in translation quality and SLU performance.

The chapter is structured as follows: we first describe the corpora used for adaptation in Section 7.2. Then we present language style adaptation for translation of speech transcriptions (Section 7.3) and domain adaptation for SMT trained on out-of-domain corpora (Section 7.4). In Section 7.5 we describe Recurrent Neural Network Language Model based re-ranking for SLU performance optimization. Section 7.6 provides concluding remarks.

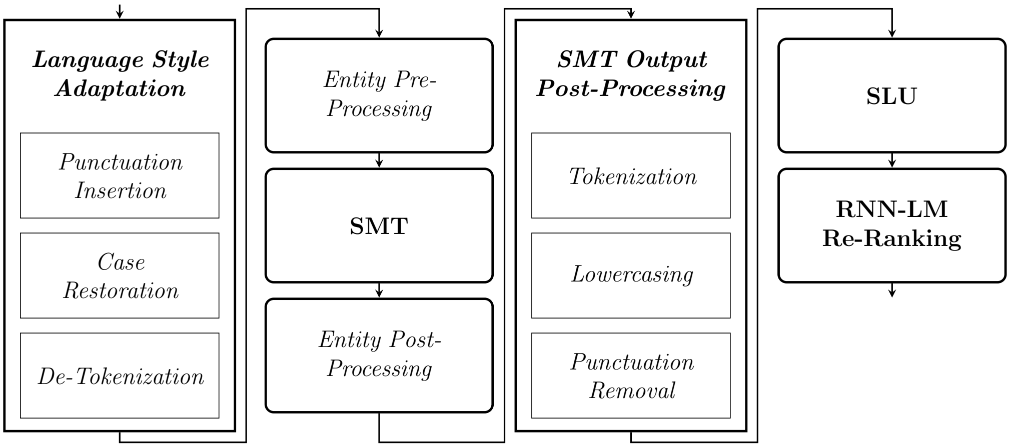


Figure 7.1: Test-on-Source Spoken Language Understanding pipeline based on Statistical Machine Translation.

7.2 Adaptation Corpora

The **in-domain** Multilingual LUNA Corpus [67] and the **out-of-domain** Europarl corpus were described in Chapters 2 and 5, respectively. In this Section we briefly describe the corpora used to adapt SMT systems.

The **close-to-domain** *LUNA Web Corpus* was crawled from the web. Starting from the original Italian LUNA corpus, rules for query construction and sentence selection were manually created. The first 100 query results returned by Google Search were downloaded. The downloaded documents were automatically sentence split, and the same handcrafted rules were used to extract sentences close to the LUNA domain. The resulting set of 80K crawled sentences was semi-automatically cleaned for encoding and spelling issues to results in a 50K sentence close-to-domain corpus.

The Spanish sentences of Europarl are additionally used to train language models for language style adaptation experiments. For Turkish, the corpus used for style adaptation is *Turkish Wikipedia* dump. The text was extracted and sentence split to result in approximately 3M sentences.

7.3 Language Style Adaptation

Using off-the-shelf SMT systems like Google Translate for SLU porting has both advantages and disadvantages. The advantages are that the SDS developers do not require expertise in machine translation, and can obtain satisfactory translations for a wide range of language pairs without the need for parallel corpora. The disadvantages, on the other hand, are that these SMT systems are general domain, and are trained on written text, which differs in style from the spoken conversation transcriptions SLU is trained on.

In this section we describe and evaluate our approach to the problem of unmatched style (see Figure 7.1). The description is organized from the SLU perspective. First, the output of SMT system is matched the conversation transcription style in the source language. Second, the input (ASR output or transcription in target language) is matched the SMT training data style, i.e. written text. Since in conversation transcription style, unlike written text, symbols and numbers are always spelled out; we apply an additional step of entity pre- and post-processing aimed at reducing the noise added by SMT.

7.3.1 SMT Output Post-Processing

The differences between conversation transcriptions (data for training SLU systems) and written text is that, the former has no sentence boundaries, no capitalization, no punctuation, and it is tokenized. Thus, SMT output post-processing step consists of tokenization, lowercasing, and removing all punctuation except single quotes used in contractions. All the techniques are widely applied in various NLP tasks.

The same process is applied to Europarl prior to training the SMT system (Europarl Moses baseline) to bring it closer to the conversational style.

7.3.2 Language Style Adaptation

The process of adapting conversation transcription style to the written text style is the direct opposite of the SMT output post-processing. Thus, the steps are: automatic punctuation insertion, automatic case restoration, and de-tokenization. De-tokenization (attaching punctuation marks and contractions to the respective tokens) is handled by language dependent rules. Case Restoration and Punctuation Insertion, on the other hand, require training of statistical models; thus, these two steps are described in more detail.

Automatic Punctuation Insertion requires language models (LM) to be trained on a lowercased and tokenized corpus that contains punctuation. The language model we use is a trigram back-off language model with modified Kneser-Ney discounting, trained on Spanish section of Europarl for Spanish-Italian translation and on Wikipedia for Turkish-Italian. Prior to training the LMs, both corpora were sentence split, and the beginning and end of sentence tags were inserted. To reduce noise we restrict the punctuation lexicon to a few most frequent punctuation marks: '.', ',', '?', '¿'. The *hidden-ngram* tool from SRILM toolkit [68], which tags a sequence of tokens with hidden events occurring between them, is used for recovering the missing punctuation.

Automatic Case Restoration requires cased corpora to train language and case models. For training these models, we use the same corpora as for automatic punctuation insertion, with the exception that the corpora remain cased. We apply *Moses recaser*, provided with the Moses translation system. The tool trains a restricted translation model to translate from lowercased to cased text. Additionally, it applies sentence initial capitalization.

7.3.3 Entity Processing for SMT

Dialogs, like any data, often contain named entities, dates, numerical expressions, etc. Moreover, all these entities are spelled out. Additionally, detection of such entities has already achieved satisfactory performance for a wide range of languages (e.g. Italian Named Entity Recognition [44]). In a live dialog system these entities are usually handled by their associated grammars, either handcrafted by the developers or provided as built-in by the ASR system. In the context of automatic translation of transcriptions or ASR output, translating such entities adds additional noise; thus, handling these entities with a grammar in the target language is a better option. The step is beneficial for both off-the-shelf and moses-based systems trained on out-of-domain corpora.

In the domain of LUNA corpus (IT Help Desk), one of the most frequent entities is numerical expressions: ticket numbers, phone numbers, etc.. Google Translate often converts word-numerical expressions into dates or reorders them. To reduce this translation noise the following procedure is implemented: (1) Each word-numerical expression in the target language (Spanish or Turkish) is converted to digits, i.e. "two thousand six" is converted to "2006". (2) In the source language (Italian) these expressions are converted back to word-numerical form, i.e. "2006" is converted to "two thousand six". The digit-form expressions are enclosed in XML tags to prevent their translation, a feature supported both by Google Translate and Moses.

7.3.4 Results and Discussion

Table 7.1 reports cumulative results of SMT output post-processing, SMT language style adaptation, and numerical entity pre- and post-processing steps for Google Translate on the LUNA development set. For output post-processing and language adaptation all the described steps are applied, since they improve BLEU score individually and in combination. Output post-processing improves the performance by almost 4 point for Spanish and 3 points for Turkish. The effect of style adaptation is greater for Spanish (2.88) than for Turkish (1.37), what is easily explained by the rich Turkish morphology; thus, greater data sparseness. Due to the high frequency of word-numerical entities in LUNA, numerical entity processing step improves the performance by additional 2.44 for Spanish and 4.31 points for Turkish. Considering all the pre- and post-processing steps, results indicate that off-the-shelf SMT systems like Google Translate can be adapted to the spoken utterance translation, and matching input and output language styles greatly improves performance irrespective of language distance (by 9.21 points for Spanish and 8.43 for Turkish).

Table 7.2 reports performance of the style-adapted SMT systems on LUNA Development and Test Sets. Affected systems are Google Translate and Europarl Moses. While the former includes the full pre- and post-processing, the latter includes only entity processing, since Europarl is adapted to conversational style prior to training. Comparing Tables 5.1 and

Pre- & Post- Processing	*Language Pair*	
	ES-IT	**TR-IT**
Baseline	25.89	13.72
+ Post-Processing	29.78	16.47
+ Style Adaptation	32.66	17.84
+ Numerical Entities	35.10	22.15

Table 7.1: Cumulative effects of output post-processing, style adaptation and numerical entity processing for Google Translate on LUNA Development Set. Results are reported as 4-gram BLEU score.

SMT System	*Language Pair*			
	ES-IT		**TR-IT**	
	DEV	**TEST**	**DEV**	**TEST**
Google Translate	35.10	31.08	22.15	20.13
Europarl Moses	37.37	35.69	N/A	N/A
LUNA Moses	49.77	50.69	33.39	35.29

Table 7.2: Performance of the style-adapted off-the-shelf SMT Google Translate, out-of-domain Europarl Moses, and in-domain LUNA Moses SMT systems on LUNA Development and Test Sets. Results are reported as 4-gram BLEU score.

7.2, we reduce the performance difference between off-the-shelf and out-of-domain Spanish - Italian SMT from 9.19 to 2.27. However, both systems still perform more than 10 points below the in-domain SMT system.

7.4 Domain Adaptation for SMT

SMT systems are drastically affected by differences in training and testing conditions. One of the drawbacks of using an off-the-shelf translation systems is not being able to access its translation and language models. Thus, any available in-domain data is not utilized. An alternative is to train the system using an open source tools such as Moses on out-of-domain parallel corpora like Europarl, and adapt it to the target domain. Thus, in this section we address the second limitation of cross-language SLU porting – scarce aligned data.

Domain Adaptation is a rather well studied topic in machine translation research, and a variety of methods were proposed (see [5] for review). Phrased-based SMT tools, like moses, generally require two models for translation: a translation model (phrase table) and a language model. With respect to the availability of bilingual in-domain data either of these models is adapted to the target domain. Simple SMT domain adaptation techniques are presented in [37]:

(1) pooling large out-of-domain and small in-domain parallel corpora together to train the models;

(2) using out-of-domain corpus for the translation model and in-domain data for the language model;

(3) their combinations;

We follow the same approach, but additionally augment the data for training the language models with close-to-domain web crawled data, i.e. LUNA Web Corpus. Thus, for Europarl Moses system, we substitute Europarl trained out-of-domain language model with a language model trained on:

(1) Italian LUNA corpus – in-domain data;

(2) LUNA Web corpus – close-to-domain data;

(3) both corpora;

In all cases monolingual target language data is used. For the sake of completeness, we also present results on pooled data training and in-domain SMT with web data augmented language model. Since Europarl is not available for Turkish, all adaptation experiments are for Spanish - Italian.

Table 7.3 reports results on domain adaptation. The first observation is that augmenting the in-domain language model with close-to-domain web crawled improves the already high performance of the LUNA Moses by 1.57 for the development and 0.08 for the test set. Using in-domain language model to re-score translation hypotheses of the out-of-domain Europarl translation model improves performance by more than 10 points. Even though the gain of using close-to-domain language model is less, the performance is still more than 8 points higher than of the out-of-domain SMT. Training the language model on both in-domain and close-to-domain corpora outperforms both and falls only 0.41 points less than the in-domain

Transl. Model	**Lang. Model**	**DEV**	**TEST**
LUNA	LUNA	49.77	50.69
LUNA	LUNA+Web LUNA	51.34	50.77
Europarl	Europarl	37.37	35.69
Europarl	LUNA	48.11	44.65
	Web LUNA	46.58	40.82
	LUNA+Web LUNA	49.36	45.60
Europarl+LUNA	Europarl+LUNA	47.57	46.87
Europarl+LUNA	Europarl+LUNA +Web LUNA	49.66	48.95

Table 7.3: Effects of domain adaptation with in-domain and close-to-domain language models for Europarl Moses Spanish-Italian SMT on LUNA Development and Test Sets. Results are reported as 4-gram BLEU score.

SMT system for the development set; however, the difference increases to 5 points on the test set.

Pooling Europarl and LUNA corpora to train translation and language models yields performance more than 10 points higher than the out-of-domain system. Augmenting the pooled data with close-to-domain data increases performance by additional 2 points, close to the in-domain SMT.

The domain adaptation experiments show that adapting out-of-domain data trained SMT systems with monolingual in-domain data and close-to-domain data yields performance close to the in-domain SMT; thus, the translation of the source language corpora to build in-domain SMT for Test-on-Source SLU might not be necessary. Augmenting the limited in-domain data with larger web-crawled close-to-domain data is definitely beneficial: the Out-Of-Vocabulary rate (OOV) for LUNA corpus drop from 4.30% to 1.27% with the addition of close-to-domain data to the training set; consequently, better performance is expected.

7.5 Test-on-Source SLU

In the Test-on-Source approach there is already an SLU model in the source language and SMT is deployed to translate the target language utterances to the source language. For the two target languages, Spanish and Turkish, utterances are translated to Italian, using the SMT systems described in the sections above. The translated utterances are the input to the SLU for semantic parsing (extraction of domain concepts).

Since the SMT systems are optimized for BLEU during training, and the target evaluation metric is CER, the behavior of the systems might be different on Spoken Language Understanding. The problem of optimizing the SMT directly for semantic parsing was addressed by tuning the moses-based SMT (setting the model weights via Minimum Error Training) in [30]. The authors showed that such tuning reduces the CER. We follow a different approach exploiting the fact that Google Translate and Moses can output several translation hypotheses (n-best list). These hypotheses are parsed by the SLU and then re-ranked using in-domain RNN-based joint LM [2] trained on reference transcription word-concept pairs.

First, we briefly describe RNN-based joint LM re-ranking, and then present the results on re-ranking of the style adapted and domain adapted SMT systems.

7.5.1 RNN-based Joint Language Model Re-Ranking

Considering word-to-concept alignment constrains to optimize language models (LMs) improves SLU performance [58]. A Neural Network (NN) LM to optimize the SLU performance, which is a joint model that is built over word-concept pairs, was proposed in [2]. The given LM is based on a recurrent NN (RNN) that uses a modified version of the class-based RNN structure given in [45]. This RNN-based joint LM is used to re-rank the n-best list of semantic parses of the translation hypotheses. Translation scores of the SMT systems are combined with the scores of the RNN-based joint LM. Specifically, translation and LM scores provided by the SMT (moses) are extracted and the LM score is substituted with the RNN-based joint LM probability. In case of Google Translate there is no separate language model score; thus, the re-ranking is solely RNN-LM score based.

7.5.2 Results and Discussion

Table 7.4 reports the SLU performance of the baseline and style-adapted systems (including all pre- and post-processing) and the in-domain LUNA

SMT System	**BL**	**SA**	**RNN-LM**
Spanish - Italian			
Google Translate	43.00	36.10	34.60 (31.10)
Europarl Moses	39.20	35.40	31.30 (22.80)
LUNA Moses	25.80	N/A	25.30 (20.70)
Turkish - Italian			
Google Translate	56.90	50.40	49.20 (44.70)
LUNA Moses	39.20	N/A	37.90 (27.70)

Table 7.4: Test-On-Source SLU performance of SMT systems on the LUNA Test Set. 1-Best SLU CER for the baseline and style-adapted systems, 100-Best RNN-LM re-ranked CER, and 100-Best oracle CER (in parentheses) are reported.

Transl. Model	**Lang. Model**	**SLU**	**RNN-LM**
LUNA	LUNA	25.80	25.30 (20.70)
LUNA	LUNA+Web LUNA	26.00	26.00 (22.80)
Europarl	Europarl	35.40	31.30 (22.80)
Europarl	LUNA	31.20	29.80 (23.60)
	Web LUNA	32.70	31.30 (25.20)
	LUNA+Web LUNA	31.20	30.00 (24.50)
Europarl+LUNA	Europarl+LUNA	28.40	27.20 (23.10)
Europarl+LUNA	Europarl+LUNA +Web LUNA	27.90	26.30 (22.10)

Table 7.5: Test-On-Source SLU performance of the domain-adapted Spanish - Italian moses SMT systems on the LUNA Test Set. 1-Best SLU CER, 100-Best RNN-LM re-ranked CER, and 100-Best oracle CER (in parentheses) are reported.

Moses SMT in terms of CER; as well as the performance of 100 best RNN-LM re-ranking (oracles of 100-best are given in parentheses). The first observation is that the performance of the systems in terms of CER is in line with their performance in terms of BLEU, i.e. in-domain SMT perform the best and the off-the-shelf SMT the worst. This holds for the baseline, style-adapted and RNN-LM re-ranked systems. Style adaptation significantly improves performance of both Google Translate and Europarl Moses. The benefits of re-ranking is greater for out-of-domain SMT than for Google Translate. This is explained by the fact that Google Translate outputs only a few translation hypotheses (on average 4.5 hypotheses per sentence), while for the moses-based systems we use 100 hypotheses. Performance improvements hold across language pairs.

SLU performance and the results of RNN-based joint LM re-ranking for domain-adapted Spanish - Italian moses-based SMT are reported in Table 7.5. Even though, in-domain SMT augmented with web crawled data has higher BLEU score (see Table 7.3), it produces worse SLU results. Similarly, for the out-of-domain SMT trained on Europarl, augmenting the in-domain LM with web crawled data does not improve SLU performance. However, for the SMT with pooled-data training, adding web crawled data to the in-domain corpus, improves performance by 0.5.

The results of 100-best RNN-LM re-ranking are in-line with 1-best SLU results for the domain adapted systems: the only benefit of adding web crawled data is observed in pooled-data training condition. The benefit of re-ranking is proportional to the amount of out-of-domain data in the language models of SMT. Thus, Europarl Moses benefits the most, CER drops by 4.1%, and reaches the performance of the SMT adapted by web-crawled data only.

7.6 Conclusion

In this chapter we proposed methods for dealing with the limitations of cross-language SLU porting such as scarceness of aligned data and unmatched style of conversation transcriptions and written text: style adaptation, domain adaptation, and semantic parse re-ranking with in-domain RNN-based LM. We evaluate end-to-end SLU system porting on both close and distant language pairs: Spanish - Italian and Turkish - Italian; and achieve significant improvements both in translation quality and SLU performance.

Chapter 8

Discourse Parsing of Conversations: Baselines and Challenges

In this Chapter we address the issues related to discourse parsing of spoken conversations. We analyze LUNA discourse annotation and present models for Discourse Connective Detection. In Chapter 4, we have demonstrated that Argument Span Extraction generalizes well across-domains. Also we have mentioned that Discourse Connective Detection, on the other hand, does not generalize well. However, it is the most critical step in discourse parsing, since the rest of the subtasks depend on it.

8.1 Introduction

The two discourse relation annotated corpora we have worked on in Chapters 3 and 4 are Penn Discourse Treebank (PDTB) [52] and Biomedical Discourse Relation Bank [54]. The both are essentially written monologues in English. Italian LUNA Corpus, on the other hand, contains discourse annotation on spoken dialogs in Italian. The issues of non-written-text and non-English discourse annotation were addressed in [70].

In this Chapter we address the issues of discourse parsing using spoken conversation corpus (LUNA). We first describe the discourse parsing differences with respect to the nature of the input: written text or speech transcription in Section 8.2. Then in Section 8.3 we analyze LUNA Corpus in terms of discourse relation annotation. In Section 8.4 we describe and evaluate Discourse Connective Detection models.

8.2 Discourse in Speech and Text

The issues of discourse relation annotation of dialogs using Rhetorical Structure Theory (RST) [69] is discussed in [63]. The main difference between written text and a spoken dialog is in their segmentation into units of a discourse relations – connective and its arguments. While for written text there is only one speaker; thus, it is straightforward; the dialog introduces an additional level of segmentation – speakers and turns. As it was mentioned in [63], discourse relations may appear cross-speaker: different arguments of the same relation being in different speaker turns for *elaboration* relation, for instance. Additionally, due to the phenomena such as one speaker completing the other's utterance, even arguments may appear cross-speaker. Overall, in spoken dialogs the turn and speaker segmentation is not parallel to the discourse relation segmentation.

The PDTB-styled discourse parser developed in Chapters 3 and 4 essentially relies on the notions of sentence and adjacency. Dialogs, on the other hand, are segmented into turns. A turn may contain a part of a sentence or one or more sentences; and this information is generally not available. Turns, on the other hand, usually consist of one or several segments, partitioned with respect to some 'event' such as short silence, speech disfluency, or other. Taking any of these notions – turn or segment – as an equivalent for a sentence is equally problematic. In Section 8.3 we analyze the LUNA discourse annotation turn-wise to assess the ratio of discourse relations that are potentially processable by a discourse parser trained on text.

8.3 Data Analysis

The Table 8.1 presents statistics on discourse relations in the LUNA Corpus. There are 1,052 explicit discourse relations in the LUNA Corpus (65.5% of total 1,606 annotated relations) which are signaled by 85 unique explicit discourse connectives. For comparison, in PDTB there are 18,459 explicit discourse relations and 100 unique explicit connectives.

In Chapter 3 we further analyzed discourse relations and connectives as inter- and intra-sentential. For LUNA Corpus, however, such analysis is not possible, since conversation transcriptions lack manual sentence

Annotation Statistic	**Counts**
Dialogs	60
Turns	3,750
Tokens	24,800
Total Relations	1,606
Explicit Relations	1,052
Unique Explicit Connectives	85
Unique Connective Surfaces	126

Table 8.1: Italian LUNA Corpus discourse annotation statistics (partially from [70]).

segmentation, and there is no reference syntactic parses. However, unlike PDTB there is a speaker and turn information. Since the discourse annotation procedure relied on the annotator's intuition for the disambiguation of overlapping turns and reconstruction of utterances, while speaker information is available in transcription layer of the corpus, we have analyzed discourse relations as *single-speaker* and *cross-speaker* relations. A discourse relation consists of three spans: connective, Argument 1 and Argument 2; thus, the analysis additionally considers single vs. cross-speaker spans. Moreover, spontaneous dialogs contain interruptions; thus, some discourse relations may lack one or both of its arguments.

The statistics of Discourse Relation Span Analysis are given in Table 8.2. Since PDTB-styled discourse parser described in Chapters 3 and 4 relies on the notion of sentence and essentially works mostly on intra-sentential discourse relations (recall that inter-sentential argument candidates are selected using heuristics), it is important to select a set of *single-speaker single-turn* relations. The ratio of such relations is only 37.6% (396), which is very low; thus, additional pre-processing for the *reconstruction* of discourse relations is required. Since such pre-processing is out of the scope of this thesis, argument span extraction for dialogs is left as a future work.

8.4 LUNA Discourse Connective Detection

Discourse connectives have shorter spans and are less affected by cross-speaker cross-turn discourse relation issues. In this section we describe the *baseline* discourse connective detection model trained on LUNA.

	Counts	%
Total	1,052	100%
Missing Span	19	1.8%
Speaker-wise Analysis		
Cross-Speaker Span	102	9.7%
Cross-Speaker Relation	170	16.2%
Single-Speaker Relation	763	72.5%
Turn-wise Analysis		
Multi-turn Span	233	(30.5%) 22.2%
Single-turn Span	530	(69.5%) 50.4%
Single-turn Relation	396	(51.9%) 37.6%

Table 8.2: Italian LUNA Corpus discourse relation span statistics. For turn-wise analysis percent of from single-speaker relations is given in parentheses.

	Training (01-02)	Testing (03)	Total
# of dialogs	48	12	60
# of explicit relations	794	258	1,052

Table 8.3: Distribution of LUNA discourse data into training and testing sets.

8.4.1 Experimental Settings

The 60 human-human dialogs of LUNA Corpus, that are annotated with discourse relation information are split into 3 sections. We use the first two sections for training and the third for testing. The distribution of data in the split is given in Table 8.3.

8.4.2 Features

The features used to train the LUNA discourse connective detection model are tokens (surface strings), part of speech tags and IOB-chains. The part-of-speech tags and IOB-chains are extracted from automatic syntactic parse trees using syntactic parser by [13]. For the experiments we considered a 'segment' to be equivalent to a sentence. The CRF [38] model is trained taking these features in the ±2 window.

Model	**P**	**R**	**F1**
LUNA: Token	64.87	28.88	40.91
LUNA: Token + POS + IOB-chain	61.96	23.65	34.23
PDTB: Token [51]	–	–	75.33
PDTB: Best [51]	–	–	94.19
PDTB - PDTB [56]	88	81	84
BioDRB - BioDRB [56]	79	63	69
PDTB - BioDRB [56]	79	42	55

Table 8.4: Discourse Connective Detection in LUNA Corpus. Results reported in terms of precision (P), recall (R) and F-measure (F1). PDTB and BioDRB in-domain and cross-domain results are given for a reference.

8.4.3 Results and Discussion

The discourse connective detection results are give in Table 8.4 for exact connective span match. For the other corpora we present published results by [51] and [56]. Their settings for PDTB are 10-fold cross-validation and for BioDRB 12-fold cross-validation. While [51] makes use of complex syntactic features extracted from gold parse trees for the best classifier, [56] makes use of tokens, n-grams and morphological information only.

The PDTB discourse connective model trained only on tokens (connective surfaces) already yields F-measure of 75.33. The LUNA model trained only on tokens yields the F-measure of 40.91. The interesting difference from the other corpora is that adding syntactic features results in a drop of performance of more than 6 points (from 40.91 to 34.23), which is indicative of the poor performance of the syntactic parser on the conversation data, as well as segments' being not appropriate for syntactic parsing. Thus, the syntactic parser should be adapted to speech data or the data should undergo the language style adaptation process of Chapter 7.

8.5 Conclusion

In this chapter we have discussed the issues related to discourse parsing of dialogs. The presented experiments on LUNA discourse connective detection indicate that small data size coupled with the nature of conversational data make the task challenging.

Chapter 9

Conclusion

In this thesis we have addressed the problems of cross-language Natural Language Processing: parallel corpora creation, domain adaptation, and language style adaptation. The problems were addressed on the tasks of Discourse Parsing and Spoken Language Understanding. The tasks are cast as token-level sequence labeling with Conditional Random Fields; thus, the majority of the proposed and evaluated techniques are applicable to both.

In Chapter 2 we have presented discourse and speech corpora used throughout the thesis. Additionally, we have presented the methodology for the creation of parallel speech corpora via professional translation services that considers speech-specific phenomena, such as disfluencies, and their replicability with respect to language distance.

One of the subtasks of Discourse Parsing is the extraction of the relation argument spans. In Chapter 3 we compare the two strategies for the argument span extraction in Penn Discourse Treebank (PDTB) style discourse relation parsing: to process intra- and inter-sentential explicit relations by a single model, or separate ones. We extend the argument span extraction approach of [22] and integrate argument position classification and immediately previous sentence heuristic. The evaluation of parsing strategies on the PDTB explicit discourse relations shows that the models trained specifically for intra- and inter-sentential relations significantly outperform the single ±2 window models of [22].

The separate model parser is further evaluated cross-domain on Biomedical Discourse Relation Bank [54] on subtasks of argument position classification and argument span extraction in Chapter 4. The observed cross-domain performances are indicative of good model generalization. However, since these models are applied later in the pipeline, they are affected by the cross-domain performance of the other tasks. Specifically, discourse connective detection, which was shown not to generalize well in the literature. Additionally, we have presented feature-level domain adaptation techniques to reduce the dependence of the cross-domain argument span extraction on the other subtask – relation sense classification .

Additionally, in Chapters 5 and 7 we have proposed methods for dealing with the limitations of cross-language system porting such as scarceness of aligned data and unmatched style of conversation transcriptions and written text: style adaptation, domain adaptation, and semantic parse re-ranking with in-domain RNN-based LM. We have evaluate end-to-end SLU system porting on both close and distant language pairs: Spanish - Italian and Turkish - Italian; and achieved significant improvements both in translation quality and SLU performance.

The intermediate tools used for cross-domain experiments are not domain adapted. Specifically, the syntactic parser (Stanford) that provides sentence splitting and tokenization is trained on Penn Treebank, i.e. it is in-domain for PDTB and out-of-domain for BioDRB; and it is known that domain-optimized tokenization improves performance on various NLP tasks. Thus, the future direction of this work is to evaluate discourse parsing using tools optimized for the domains in question.

One particular property of human-human dialogs is that there is no sense of a sentence and many overlapping turns. The discourse parsing, however, relies on this information. In Chapter 8, we have presented LUNA Corpus analysis with respect this notions and discussed the related challenges; and the baseline results.

To sum up, even though cross-language and cross-domain porting of shallow parsing applications – Discourse Parsing and Spoken Language Understanding – is successful in isolation, cross-style application is still challenging.

Bibliography

[1] Collin F. Baker, Charles J. Fillmore, and John B. Lowe. The berkeley framenet project. In *Proceedings of the 34th Meeting of the Association for Computational Linguistics (ACL) / the 17th International Conference on Computational Linguistics (COLING)*, 1998.

[2] Ali Orkan Bayer and Giuseppe Riccardi. Joint language models for automatic speech recognition and understanding. In *Proceeding of the IEEE Spoken Language Technology Workshop*, 2012.

[3] Ali Orkan Bayer and Giuseppe Riccardi. On-line adaptation of semantic models for spoken language understanding. In *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding*, 2013.

[4] Luisa Bentivogli, Pamela Forner, and Emanuele Pianta. Evaluating cross-language annotation transfer in the multisemcor corpus. In *Proceedings of the 20th International Conference on Computational Linguistics*. Association for Computational Linguistics, 2004.

[5] Nicola Bertoldi and Marcelo Federico. Domain adaptation for statistical machine translation with monolingual resources. In *Proceedings of the Workshop on Statistical Machine Translation*, 2009.

[6] Cornelia Brunner and Thomas Wirth. Btk expression is controlled by oct and bob. 1/obf. 1. *Nucleic acids research*, 34(6):1807–1815, 2006.

[7] Sabine Buchholz. Readme for perl script chunklink.pl, 2000.

[8] Harry Bunt. A framework for dialogue act specification. In *In Proceedings of SIGSEM WG on Representation of Multimodal Semantic Information*, 2005.

[9] Chris Callison-Burch and Mark Dredze. 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*, pages 1–12. Association for Computational Linguistics, 2010.

[10] CDCOLOGY. http://www.cdcology.sparked.com/, March 2014.

[11] Timothy Chklovski and Rada Mihalcea. Building a sense tagged corpus with open mind word expert. In *Proceedings of the ACL-02 workshop on Word sense disambiguation: recent successes and future directions-Volume 8*, pages 116–122. Association for Computational Linguistics, 2002.

[12] Shammur Absar Chowdhury, Arindam Ghosh, Evgeny A. Stepanov, Ali Orkan Bayer, Giuseppe Riccardi, and Ioannis Klasinas. Cross-language transfer of semantic annotation via targeted crowdsourcing. In *Proceedings of the 15th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, Singapore, September 2014.

[13] Anna Corazza, Alberto Lavelli, and Giorgio Satta. Analisi sintattica-statistica basata su costituenti. *Intelligenza Artificiale*, 4(2):38–39, 2007.

[14] Mark G. Core and James F. Allen. Coding dialogs with the damsl annotation scheme. In *Proceedings of AAAI Fall Symposium on Communicative Action in Humans and Machines*, 1997.

[15] Marco Dinarelli, Silvia Quarteroni, Sara Tonelli, Alessandro Moschitti, and Giuseppe Riccardi. Annotating spoken dialogs: from speech segments to dialog acts and frame semantics. In *Proceedings of EACL Workshop on the Semantic Representation of Spoken Language*, Athens, Greece, 2009.

[16] Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Rashmi Prasad, Aravind Joshi, and Bonnie Webber. Attribution and the (non)-alignment of syntactic and discourse arguments of connectives. In *Proceedings of*

the ACL Workshop on Frontiers in Corpus Annotation II: Pie in the Sky, 2005.

[17] Steven Dow, Anand Kulkarni, Brie Bunge, Truc Nguyen, Scott Klemmer, and Björn Hartmann. Shepherding the crowd: managing and providing feedback to crowd workers. In *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, pages 1669–1674. ACM, 2011.

[18] Robert Elwell and Jason Baldridge. Discourse connective argument identification with connective specific rankers. In *Proceedings of the IEEE International Conference on Semantic Computing (ICSC 2008)*, 2008.

[19] Syeed Ibn Faiz and Robert E Mercer. Identifying explicit discourse connectives in text. In *Advances in Artificial Intelligence*, pages 64–76. Springer, 2013.

[20] Tim Finin, Will Murnane, Anand Karandikar, Nicholas Keller, Justin Martineau, and Mark Dredze. Annotating named entities in twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 80–88. Association for Computational Linguistics, 2010.

[21] Joseph L. Fleiss. Measuring agreement between two judges on the presence or absence of a trait. *Biometrics*, 31:651–659, 1975.

[22] Sucheta Ghosh, Richard Johansson, Giuseppe Riccardi, and Sara Tonelli. Shallow discourse parsing with conditional random fields. In *Proceedings of the 5th International Joint Conference on Natural Language Processing (IJCNLP 2011)*, 2011.

[23] Sucheta Ghosh, Richard Johansson, Giuseppe Riccardi, and Sara Tonelli. Improving the recall of a discourse parser by constraint-based postprocessing. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*, 2012.

[24] Sucheta Ghosh, Giuseppe Riccardi, and Richard Johansson. Global features for shallow discourse parsing. In *Proceedings of the SIGDIAL 2012 Conference, The 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 150–159, 2012.

[25] Xiaodong He, Li Deng, Dilek Hakkani-Tür, and Gokhan Tur. Multi-style adaptive training for robust cross-lingual spoken language understanding. In *Proceedings of the ICASSP 2013, IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2013.

[26] George Hripcsak and Adam S Rothschild. Agreement, the f-measure, and reliability in information retrieval. *Journal of the American Medical Informatics Association*, 12(3):296–298, 2005.

[27] Pei-Yun Hsueh, Prem Melville, and Vikas Sindhwani. Data quality from crowdsourcing: a study of annotation selection criteria. In *Proceedings of the NAACL HLT 2009 workshop on active learning for natural language processing*, pages 27–35. Association for Computational Linguistics, 2009.

[28] Rebecca Hwa, Philip Resnik, Amy Weinberg, and Okan Kolak. Evaluating translational correspondence using annotation projection. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 392–399. Association for Computational Linguistics, 2002.

[29] Bassam Jabaian, Laurent Besacier, and Fabrice Lefèvre. Investigating multiple approaches for SLU portability to a new language. In *Proceedings of INTERSPEECH*, 2010.

[30] Bassam Jabaian, Laurent Besacier, and Fabrice Lefèvre. Combination of stochastic understanding and machine translation systems for language portability of dialogue systems. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2011.

[31] Bassam Jabaian, Laurent Besacier, and Fabrice Lefèvre. Comparison and combination of lightly supervised approaches for language porta-

bility of a spoken language understanding system. *IEEE Transactions on Audio, Speech, and Language Processing*, 21(3), 2013.

[32] Richard Johansson and Alessandro Moschitti. Syntactic and semantic structure for opinion expression detection. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 67–76, 2010.

[33] J-D Kim, Tomoko Ohta, Yuka Tateisi, and Jun'ichi Tsujii. Genia corpus—a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(suppl 1):i180–i182, 2003.

[34] Dan Klein and Christopher D. Manning. Fast exact inference with a factored model for natural language parsing. In *Advances in Neural Information Processing Systems 15 (NIPS 2002)*, pages 3–10, 2003.

[35] Alistair Knott. *A Data-Driven Methodology for Motivating a Set of Coherence Relations.* PhD thesis, University of Edinburgh, 1996.

[36] Philipp Koehn. Europarl: A parallel corpus for statistical machine translation. In *Machine Translation Summit X*, 2005.

[37] Philipp Koehn and Josh Schroeder. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, StatMT '07, pages 224–227, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics.

[38] John Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of 18th International Conference on Machine Learning*, pages 282–289, 2001.

[39] Geoffrey Leech and Andrew Wilson. Eagles: Recommendations for the morphosyntactic annotation of corpora, 1996.

[40] Fabrice Lefèvre, François Mairesse, and Steve Young. Cross-lingual spoken language understanding from unaligned data using discriminative classification models and machine translation. In *Proceedings of INTERSPEECH*, 2010.

[41] Fabrice Lefèvre, Djamel Mostefa, Laurent Besacier, Yannick Estève, Matthieu Quignard, Nathalie Camelin, Benoit Favre, Bassam Jabaian, and Lina M. Rojas-Barahona. Leveraging study of robustness and portability of spoken language understanding systems across languages and domains: the portmedia corpora. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, 2012.

[42] Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. A pdtb-styled end-to-end discourse parser. *Natural Language Engineering*, 1:1 – 35, 2012.

[43] Matthew Marge, Satanjeev Banerjee, and Alexander I Rudnicky. Using the amazon mechanical turk for transcription of spoken language. In *2010 IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pages 5270–5273. IEEE, 2010.

[44] Yashar Mehdad, Vitalie Scurtu, and Evgeny A. Stepanov. Italian named entity recognizer participation in ner task@ evalita 09. *Proceedings of EVALITA (Evaluation of Natural Language and Speech Tools for Italian)*, 2009.

[45] Tomáš Mikolov, Stefan Kombrink, Lukáš Burget, Jan "Honza" Černocký, and Sanjeev Khudanpur. Extensions of recurrent neural network language model. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2011.

[46] Guido Minnen, John Carroll, and Darren Pearce. Applied morphological processing of english. *Natural Language Engineering*, 2001.

[47] Matteo Negri and Yashar 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. Association for Computational Linguistics, 2010.

[48] Sebastian Padó and Mirella Lapata. Cross-lingual annotation projection for semantic roles. *Journal of Artificial Intelligence Research*, 36(1):307–340, 2009.

[49] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 2002.

[50] Gabriel Parent and Maxine Eskenazi. Toward better crowdsourced transcription: Transcription of a year of the let's go bus information system data. In *IEEE Spoken Language Technology Workshop (SLT)*, pages 312–317. IEEE, 2010.

[51] Emily Pitler and Ani Nenkova. Using syntax to disambiguate explicit discourse connectives in text. In *Proceedings of the ACL-IJCNLP Conference*, 2009.

[52] Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. The penn discourse treebank 2.0. In *Proceedings of the 6th International Conference on Language Resources and Evaluation (LREC 2008)*, 2008.

[53] Rashmi Prasad, Aravind Joshi, and Bonnie Webber. Exploiting scope for shallow discourse parsing. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC-2010)*, 2010.

[54] Rashmi Prasad, Susan McRoy, Nadya Frid, Aravind Joshi, and Hong Yu. The biomedical discourse relation bank. *BMC Bioinformatics*, 12(1):188, 2011.

[55] Balaji Polepalli Ramesh, Rashmi Prasad, Tim Miller, Brian Harrington, and Hong Yu. Automatic discourse connective detection in biomedical text. *Journal of the American Medical Informatics Association*, 19(5):800–808, 2012.

[56] Balaji Polepalli Ramesh and Hong Yu. Identifying discourse connectives in biomedical text. *AMIA Annual Symposium Proceedings*, 2010:657, 2010.

[57] Giuseppe Riccardi, Arindam Ghosh, Shammur Absar Chowdhury, and Ali Orkan Bayer. Motivational feedback in crowdsourcing: A case study in speech transcription. In *Proceedings of the INTERSPEECH*, pages 1111–1115, Lyon, France, August 2013.

[58] Giuseppe Riccardi and Allen L. Gorin. Stochastic language models for speech recognition and understanding. In *Proceedings of the International Conference on Spoken Language Processing*, 1998.

[59] Ellen Riloff, Charles Schafer, and David Yarowsky. Inducing information extraction systems for new languages via cross-language projection. In *Proceedings of the 19th International Conference on Computational Linguistics - Volume 1*, pages 1–7. Association for Computational Linguistics, 2002.

[60] Joel Ross, Andrew Zaldivar, Lilly Irani, and Bill Tomlinson. Who are the turkers? worker demographics in amazon mechanical turk. *Department of Informatics, University of California, Irvine, USA, Tech. Rep*, 2009.

[61] Robert E. Schapire and Yoram Singer. Boostexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000.

[62] Kathrin Spreyer and Anette Frank. Projection-based acquisition of a temporal labeller. In *Proceedings of the International Joint Conference on Natural Language Processing*, pages 489–496, 2008.

[63] Amanda Stent. Rhetorical structure in dialog. In *Proceedings of the first international conference on Natural language generation-Volume 14*, pages 247–252. Association for Computational Linguistics, 2000.

[64] Evgeny A. Stepanov, Ilya Kashkarev, Ali Orkan Bayer, Giuseppe Riccardi, and Arindam Ghosh. Language style and domain adaptation

for cross-language slu porting. In *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 144–149, Olomouc, Czech Republic, December 2013. IEEE.

[65] Evgeny A. Stepanov and Giuseppe Riccardi. Comparative evaluation of argument extraction algorithms in discourse relation parsing. In *13th International Conference on Parsing Technologies (IWPT 2013)*, pages 36–44, Nara, Japan, November 2013.

[66] Evgeny A. Stepanov and Giuseppe Riccardi. Towards cross-domain pdtb-style discourse parsing. In *EACL Workshops - Louhi 2014: The Fifth International Workshop on Health Text Mining and Information Analysis*, Gothenburg, Sweden, April 2014.

[67] Evgeny A. Stepanov, Giuseppe Riccardi, and Ali Orkan Bayer. The development of the multilingual luna corpus for spoken language system porting. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 2675–2678, Reykjavik, Iceland, May 2014.

[68] Andreas Stolcke. SRILM - an extensible language modeling toolkit. In *Proceedings of INTERSPEECH*, 2002.

[69] Maite Taboada and William C. Mann. Applications of rhetorical structure theory. *Discourse Studies*, 8(4):567–88, 2006.

[70] Sara Tonelli, Giuseppe Riccardi, Rashmi Prasad, and Aravind K. Joshi. Annotation of discourse relations for conversational spoken dialogs. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, 2010.

[71] David Traum. Conversational agency: The trains-93 dialogue manager. In *Proceedings of Twente Workshop on Language Technology, TWLT-II*, 1996.

[72] Maja Vukovic and Arjun Natarajan. Operational excellence in it services using enterprise crowdsourcing. In *Services Computing (SCC), 2013 IEEE International Conference on*, pages 494–501. IEEE, 2013.

[73] Bonnie L. Webber, Markus Egg, and Valia Kordoni. Discourse structure and language technology. *Natural Language Engineering*, pages 1–54, 2011.

[74] Ben Wellner and James Pustejovsky. Automatically identifying the arguments of discourse connectives. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL 2007)*, 2007.

[75] Chenhai Xi and Rebecca Hwa. A backoff model for bootstrapping resources for non-english languages. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 851–858. Association for Computational Linguistics, 2005.

[76] Fan Xu, Qiao Ming Zhu, and Guo Dong Zhou. A unified framework for discourse argument identification via shallow semantic parsing. In *Proceedings of 24th International Conference on Computational Linguistics (COLING 2012): Posters*, 2012.

[77] Hiroyasu Yamada and Yuji Matsumoto. Statistical dependency analysis with support vector machines. In *Proceedings of 8th International Workshop on Parsing Technologies*, 2003.

[78] David Yarowsky, Grace Ngai, and Richard Wicentowski. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the First International Conference on Human Language Technology Research*, pages 1–8. Association for Computational Linguistics, 2001.

[79] Omar F. Zaidan and Chris Callison-Burch. Crowdsourcing translation: Professional quality from non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 1220–1229. Association for Computational Linguistics, 2011.

Appendix A

Extracting and Using Attribution

In Chapter 2 it was mentioned that besides the discourse connective and its arguments PDTB contains annotation for attributions of discourse relations. Extraction of attribution spans is an important task, since, as noted in [16], the attribution is the main difference between syntactic and discourse argument spans. Consequently, the task has a potential to positively affect Argument Span Extraction performance.

In this Appendix, we first test the utility of the attribution for Argument Span Extraction. Then train and test automatic Attribution Span Extraction models using the features that were used for Argument Span Extraction. Finally, we evaluate the effect of automatic attribution spans on Argument Span Extraction.

A.1 Attribution as a Feature for Argument Span Extraction

Prior to training automatic Attribution Span Extraction models, it makes sense to assess the usefulness of this task for Argument Span Extraction. For this purpose, 'gold' attribution spans are used as a feature for Argument Span Extraction of intra-sentential (i.e. both *Arg1* and *Arg2* appear in the same sentence) explicit discourse relations in PDTB exactly the same way as a *Arg2* feature is used for *Arg1* span extraction. The results for this setting, presented in Table A.1, support the hypothesis. Even though the span extraction of both *Arg1* and *Arg2* is affected positively, the effect is larger for the *Arg1*: +4.53 for *Arg1* and +0.13 for *Arg2* in f-measure.

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
No Attribution	89.28	88.00	88.64	70.38	68.00	69.17
Gold Attribution	89.56	88.00	88.77	75.03	72.71	73.70

Table A.1: Argument span extraction performance of Separate Model Parser on PDTB intra-sentential relations (SS case) using 'gold' attribution spans as a feature. Results are reported as precision (P), recall (R) and F-measure (F1) without error propagation from previous steps.

Model	**P**	**R**	**F1**
TOK	63.04	46.26	53.36
TOK + POS	59.35	47.84	52.98
TOK + POS + IOB	61.67	55.05	58.17
POS	63.57	21.49	32.13
POS + IOB	61.42	51.11	55.79

Table A.2: Attribution span extraction performance on PDTB using different feature combinations. Results are reported as precision (P), recall (R) and F-measure (F1).

A.2 Automatic Attribution Span Extraction

In [42], a MaxEnt model for automatic extraction of attribution spans was trained on PDTB Sections 02-21 and tested on Section 23 using. The authors see attribution extraction as a two step process: (1) segmentation of sentences into clauses, and (2) classification of these clauses as attribution or not. The exact match results using 'gold' features in their setting have f-measure of 65.95. Unfortunately, the authors do not report using automatic attribution spans as a feature for argument span extraction.

We, on the other hand, approach Attribution Span Extraction as the token-level sequence labeling with CRFs, i.e. the same approach as for Argument Span Extraction. We train CRFs on PDTB Sections 02-22 and test on Section 23-24 using combinations of the basic features: tokens (*TOK*), part-of-speech tags (POS) and IOB-chains (*IOB*). The results are reported in Table A.2, as well as the results of [42], indicate that Attribution Span Extraction is a difficult task. The best performing CRF model that makes use of the three features has f-measure of only 58.17. Using additional features and clause segmentation might improve the performance; however, they are out of the scope of this work.

	Arg1		
	P	**R**	**F1**
No Attribution	70.38	68.00	69.17
Gold Attribution	75.03	72.71	73.70
Auto Attribution	66.60	64.41	65.48

Table A.3: Argument span extraction performance of Separate Model Parser on PDTB intra-sentential relations (SS case) using automatic attribution spans as a feature. Results are reported as precision (P), recall (R) and F-measure (F1) without error propagation from previous steps.

A.3 Argument Span Extraction with Automatic Attribution Spans

Since the Attribution Span Extraction has a low performance, we do not expect it to have a positive effect on the Argument Span Extraction performance. The reason for this is that the evaluation settings are strict and consider only exact matches to be correct; thus, using automatic attribution spans as a feature should yield more error.

Since 'gold' attribution spans have a significant positive effect only for *Arg1* span extraction, the evaluation is only for *Arg1* of intra-sentential explicit relations. The results presented in Table A.3 confirm our expectations: using the output of the Attribution Span Extraction model trained on token, part-of-speech tag and IOB-chain (TOK + POS + IOB in Table A.2) the f-measure drops 3.69 points in comparison to the models that do not make use of the feature (i.e. *No Attribution*).

A.4 Conclusion and Future Work

We have demonstrated that using additional spans as a feature for *Arg1* span extraction improves the performance. However, Attribution Span Extraction is challenging on its own and yields a performance too low to be useful. Consequently, the future directions of this work is to experiments with different features and the clause segmentation for the Attribution Span Extraction.

Appendix B

PDTB Supplementary Argument Spans as Partial Match Measure

Penn Discourse Treebank [52] follows the 'minimality principle' in the annotation of argument spans, and only portions of text minimally necessary for the interpretation of the discourse relation are included into the span. Any other span that is relevant but not minimally necessary is annotated as *supplementary information*. Following argument naming conventions, a text span supplementary to *Arg1* is labeled as *Sup1*, and to *Arg2* and *Sup2*. Additionally, besides the *exact match* evaluation, followed in this thesis, the researches make use of *partial match* evaluation (e.g. [22, 42]). However, this *partial match* metric is defined differently. In this Appendix we define an alternative *partial match* metric making use of the *supplementary* argument spans. Specifically, we evaluate Argument Span Extraction allowing variability with respect to these segments of text.

B.1 Argument Span Extraction Evaluation with Supplementary Span Variability

In PDTB, *Sup1* and *Sup2* sometimes overlap with the spans of *Arg1* and *Arg2*. Consequently, we evaluate argument spans in four conditions:

- *ARG* – the span annotated as an argument, i.e. the evaluation of Chapters 3 and 4;
- *ARG-SUP* – the difference between argument and supplementary in-

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
ARG	89.28	88.00	88.64	70.38	68.00	69.17
ARG-SUP	88.66	87.38	88.02	69.43	67.08	68.23
ARG+SUP	89.80	88.51	89.15	69.96	67.59	68.75
ANY	90.53	89.23	89.88	71.34	68.92	70.11

Table B.1: Argument span extraction performance of Separate Model Parser on PDTB intra-sentential relations (SS case) allowing variability in Supplementary Information spans. Results are reported as precision (P), recall (R) and F-measure (F1) without error propagation from previous steps.

formation spans, i.e. overlapping supplementary span is removed from the argument span;

- *ARG+SUP* – the supplementary span is joined with the argument span;
- *ANY* – any of the above is considered correct span;

The Separate Model parser models trained in Chapter 3 are evaluated with these new reference spans (as a reminder: Sections 02-22 are used for training and Sections 23-24 for testing). However, similar to the Attribution Span Extraction experiments, we do not propagate error from the Argument Position Classification and *Arg2* span extraction steps.

The results reported in Table B.1 indicate that both removing or adding Supplementary Information spans to the arguments spans do not produce significant effect on the performance. However, with the exception of *Arg2* span being joined with the *Sup2*, the effect is negative. Allowing the variability, on the other hand (*ANY* condition), has a positive effect on performance: 0.94 for *Arg1* and 1.24 for *Arg2* in f-measure.

B.2 Argument Span Extraction Evaluation with Attribution and Supplementary Span Variability

Allowing variability in argument span with respect to supplementary information is a more relaxed form of evaluation that yields better performance. Moreover, the evaluation is supported by the annotation. In order to assess

	Arg2			**Arg1**		
	P	**R**	**F1**	**P**	**R**	**F1**
ARG	89.56	88.00	88.77	75.03	72.41	73.70
ARG-SUP	88.94	87.38	88.15	74.60	71.59	72.86
ARG+SUP	90.08	88.51	89.29	74.18	72.00	73.28
ANY	90.81	89.23	90.02	75.88	73.23	74.53

Table B.2: Argument span extraction performance of Separate Model Parser on PDTB intra-sentential relations (SS case) with 'gold' Attribution Spans and allowing variability in Supplementary Information spans. Results are reported as precision (P), recall (R) and F-measure (F1) without error propagation from previous steps.

the upper bound of the Argument Span Extraction models on PDTB intra-sentential explicit discourse relations, we additionally evaluate the models previously trained and tested with 'gold' attribution spans (see Appendix A).

The results given in Table B.2 suggest that with the current discourse parser architecture and the features (including attribution spans), *Arg1* spans can be extracted with the f-measure of 74.52, and *Arg2* spans with the f-measure of 90.02. Unfortunately, this is yet far from being possible due to the low Attribution Span Extraction performance.

B.3 Conclusion

In this Appendix we have presented experiments on using Supplementary Information spans to define a partial match metric for the Argument Span Extraction task. Additionally, we have tested the approach incorporating the 'gold' attribution span feature to establish an upper bound Argument Span Extraction performance for the current state of the discourse parser cast as token-level sequence labeling with CRFs and our feature set.