Learning of text-level discourse parsing

A dissertation presented

by

Gregor Weiss

to

The Faculty of Computer and Information Science

in partial fulfilment of the requirements for the degree of

Doctor of Science

in

Computer and Information Science

Ljubljana, 2019

APPROVAL

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgement has been made in the text.

— Gregor Weiss —
May 2019

The submission has been approved by

dr. Marko Bajec
Professor of Computer and Information Science
ADVISOR

dr. Igor Kononenko
Professor of Computer and Information Science
EXAMINER

dr. Janez Demšar
Professor of Computer and Information Science
EXAMINER

dr. Flavius Frasincar
Assistant Professor of Information Systems
EXTERNAL EXAMINER
Erasmus University Rotterdam, Netherlands

PREVIOUS PUBLICATION

I hereby declare that the research reported herein was previously published/submitted for publication in peer reviewed journals or publicly presented at the following occasions:

[1] Weiss G (2015) Learning Representations for Text-level Discourse Parsing. In *Proceedings of the ACL-IJCNLP 2015 Student Research Workshop*, edited by K Chen et al.. DOI: 10.3115/v1/P15-3003.

[2] Weiss G & Bajec M (2016) Discourse Sense Classification from Scratch using Focused RNNs. In *Proceedings of the CoNLL-16 Shared Task*, edited by N Xue. DOI: 10.18653/v1/K16-2006.

[3] Weiss G & Bajec M (2018) Sense classification of shallow discourse relations with focused RNNs. *PLOS ONE*. DOI: 10.1371/journal.pone.0206057.

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my thesis. I certify that the above material describes work completed during my registration as graduate student at the University of Ljubljana.

Univerza *v Ljubljani*
Fakulteta *za računalništvo in informatiko*

Gregor Weiss
Učenje besedilnega razčlenjevanja diskurzov

POVZETEK

Razumevanje smisla diskurznih relacij, ki nastopajo med segmenti besedila, je ključnega pomena za razumevanje kateregakoli besedila v naravnem jeziku. Številni avtomatizirani pristopi so že bili predlagani, vendar so vsi odvisni od zunanjih virov, ročno-idelanih značilk in njihovi cevovodi za procesiranje so izdelani iz bistveno različnih modelov. Namesto izdelave sistema specializiranega za dani jezik in nalogo, mi stremimo k jezikovno-neodvisnemu pristopu za klasifikacijo smisla v plitkih diskurznih relacijah.

V pričujoči disertaciji najprej predstavimo naše osredotočene rekurentne nevronske mreže (focused RNNs), ki predstavljajo prvi več-dimenzionalni RNN-pozornostni mehanizem za izdelavo vložitev stavkov/argumentov. Sestavljen je iz filtrirnega RNN z mehanizmom za filtriranje/usmerjanje, ki omogoča sledečim RNN-jem, da se osredotočijo na različne vidike vsakega argumenta diskuzne relacije in ga projecirajo v več vložitvenih podprostorov. Omenjeni mehanizem uporabimo v našem sistemu FR system, ki predstavlja novo metodo za klasifikacijo smisla v plitkih diskurznih relacijah. V nasprotju z obstoječimi sistemi je FR system sestavljen iz enega modela, ki ga je mogoče celostno učiti od začetka-do-kraja, obravnava vse vrste in specifične situacije v diskurznih relacijah, ne potrebuje ročno-izdelanih značilk ali zunanjih virov, se lahko skorajda brez sprememb uporabi na kateremkoli jeziku ali oznakah smisla, in se lahko uporablja tako na ravni besed kot na ravni znakov.

Predlagani FR system smo ovrednotili na uradnih podatkovnih zbirkah in po metodologiji izziva CoNLL 2016 Shared Task. Ne zaostaja veliko za najuspešnejšimi sistemi na angleškem jeziku, vendar presega ostale sisteme brez focused RNNs plasti za 8% na kitajski podatkovni zbirki. Nato smo izvedli natančnejšo analizo na obeh jezikih.

Ključne besede procesiranje naravnega jezika, plitke diskurzne relacije, rekurentne nevronske mreže, mehanizmi pozornosti, jezikovna neodvisnost, brez zunanjih virov

University *of Ljubljana*
Faculty *of Computer and Information Science*
Gregor Weiss
Learning of text-level discourse parsing

ABSTRACT

Understanding the sense of discourse relations that appear between segments of text is essential to truly comprehend any natural language text. Several automated approaches have been suggested, but all rely on external resources, linguistic feature engineering, and their processing pipelines are built from substantially different models. Instead of designing a system specifically for a given language and task, we pursue a language-independent approach for sense classification of shallow discourse relations.

In this dissertation we first present our focused recurrent neural networks (focused RNNs) layer, the first multi-dimensional RNN-attention mechanism for constructing sentence/argument embeddings. It consists of a filtering RNN with a filtering/gating mechanism that enables downstream RNNs to focus on different aspects of each argument of a discourse relation and project it into several embedding subspaces. On top of the proposed mechanism we build our FR system, a novel method for sense classification of shallow discourse relations. In contrast to existing systems, the FR system consists of a single end-to-end trainable model for handling all types and specific situations of discourse relations, requires no feature engineering or external resources, can be used almost out-of-the-box on any language or set of sense labels, and can be applied at the word and character level representation.

We evaluate the proposed FR system using the official datasets and methodology of CoNLL 2016 Shared Task. It does not fall a lot behind state-of-the-art performance on English, but it outperforms other systems without a focused RNNs layer by 8% on the Chinese dataset. Afterwards we perform a detailed analysis on both languages.

Keywords natural language processing, shallow discourse relations, recurrent neural networks, attention mechanisms, language-independent, no external resources

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the help and support from many people. I would like to thank my advisor dr. Marko Bajec for letting me explore my interests and providing remarks on my work. I thank my committee, dr. Igor Kononenko, dr. Janez Demšar, and dr. Flavius Frasincar, for the valuable feedback and insightful comments that certainly improved this dissertation. I am grateful to all the supportive colleagues at the Laboratory for data technologies for keeping company, their sense of humor, and all the endless discussions. A great thanks goes to my parents for motivating my scientific curiosity and supporting me throughout my life. Last but not least, with all my heart I want to thank my beloved Amela for being there for me and staying positive even in the hardest times. I love you.

— Gregor Weiss, Ljubljana, May 2019.

CONTENTS

Povzetek *vii*

Abstract *ix*

Acknowledgements *xi*

1 Introduction *1*

1.1 Motivation . . . 2
1.2 Problem definition . . . 5
1.3 Scientific contributions . . . 7
1.4 Dissertation overview . . . 8

2 Background *11*

2.1 Shallow discourse relations . . . 12
2.1.1 Chinese datasets . . . 15
2.1.2 English datasets . . . 17
2.2 Existing systems . . . 20
2.2.1 Sense classifiers for `Explicit` relations . . . 21
2.2.2 Sense classifiers for `Implicit` relations . . . 23
2.3 Deep neural networks . . . 24

3 Multi-dimensional RNN-attention mechanism (focused RNNs) *27*

3.1 Sentence/argument embeddings . . . 28
3.2 Focused RNNs layer . . . 31
3.2.1 Filtering RNN . . . 33
3.2.2 Filtering/gating mechanism . . . 33

3.2.3 Focused downstream RNNs 34

4 Language-independent method for sense classification (FR system) *35*
4.1 Our approach 36
4.2 FR system 37
4.2.1 Word or character embedding layer 39
4.2.2 Focused RNNs layer 41
4.2.3 Feed-forward classification layer 42
4.3 Implementation of the FR system 43
4.3.1 Data augmentation 44
4.3.2 Hyper-parameters 45

5 Evaluation *49*
5.1 Methodology 50
5.2 Results on Chinese 53
5.2.1 Analysis of `Explicit` relations 57
5.2.2 Analysis of non-`Explicit` relations 60
5.3 Results on English 63
5.3.1 Analysis of `Explicit` relations 67
5.3.2 Analysis of non-`Explicit` relations 70
5.3.3 Case study of particular errors 73
5.4 Time complexity 77

6 Conclusion *79*
6.1 Scientific contributions 80
6.2 Future directions 81
6.3 Concluding remarks 84

Bibliography *87*

Abbreviations *91*

Razširjeni povzetek *93*

I

Introduction

1.1 Motivation

One could say that our capacity for symbolic thought and communication makes us human. It allows us to use sounds, drawings, or letters to discuss things, abstract ideas, and relations between them, even if they are not right in front of us. Any healthy human child effortlessly learns a language without explicit teaching, on the basis of positive evidence or samples, and in a *language-independent* way [1]. This happens in absence of explicitly given structure of a sentence, rules of grammar, lexicons of words, descriptions of their meaning, and relations between them. They are also not given a set of *hand-engineered features* or patterns to pay attention to, or a complex pipeline of processing steps adapted for each type of conversation or task. It all just happens naturally. On the contrary, computational approaches for many natural language processing (NLP) tasks rely on all these and are designed for a specific language and task. They process information in complex pipelines consisting of substantially different NLP components and using external resources. Such systems can not be trained in an end-to-end manner or easily adapt to new samples. These limitations are even more apparent for non-English languages, where progress in NLP is still lacking. We argue that there must be a better way, that even computational methods could learn complex NLP tasks in a language-independent way from samples alone, just like children do.

Any natural language text means more than the sum of its pieces and sentences [2]. Consequently, in order to truly comprehend a text, we need to identify its pieces and infer additional semantic relations between them, known as *discourse relations* or coherence relations. Although discourse relations are an important part of each language, we still do not understand this phenomena perfectly. Discourse relations have long been recognized as important for many structure-enabled NLP applications [3], such as text summarization [4], information extraction [5], statistical machine translation [6], sentiment analysis [7], question generation [8], coherence modelling [9], and text-level discourse parsing [10].

Over the last three decades, linguists proposed a number of theoretical text-level discourse frameworks to describe the language at the sentence level and at the text level [2, 11]. In this dissertation we have chosen the text-level discourse framework of *shallow discourse relations*, also called PDTB-style, because it strives to maintain a theory-neutral approach and offers the largest annotated corpora [12]. It defines a *discourse argument* (`arg1`, `arg2`) as a piece of text meant to communicate specific information (abstract

objects, events, states, facts, or propositions). And a *discourse relation* as a semantic relation that describes how a pair of discourse arguments are related to each other and which meaning or *sense label* we infer from it. Sometimes discourse relations are explicitly formulated by using connectives (conn, e.g. *while*, *but*, *unless*) or indicated by punctuation (punc), but often identified by the reader who tries to make sense of the text. See Section 2.1 for more details about this. Let us examine a few illustrative examples:

1. *[Jane fell over]*$_{arg1}$, *while* *[Tarzan helped her]*$_{arg2}$.
2. *[I want to go to China]*$_{arg1}$, *but* *[I prefer clean air]*$_{arg2}$.
3. 除非 *[* 火车 晚点 *]*$_{arg1}$ ， 否则 *[* 我 会在 九点 钟 到 那里 *]*$_{arg2}$ 。
   (*Unless* *[the train is late]*$_{arg1}$, *(otherwise)* *[I will be there at nine o'clock]*$_{arg2}$.)

In the above examples each discourse relation contains an explicit discourse connective, hence they are called `Explicit` relations. For humans it is easy to identify the discourse connective and the meaning or sense of the discourse relation it signals. In example 1. *while* signals a temporal synchrony of two events, in 2. *but* signals a contrast, and in 3. *unless* signals an alternative outcome. From a computational perspective, it is relatively straightforward to predict these senses by carefully designing production rules to only disambiguate the function of discourse connectives [13].

1. *[Jane fell over]*$_{arg1}$. *[Tarzan helped her]*$_{arg2}$.
2. *[I want to go to China]*$_{arg1}$. *[I prefer clean air]*$_{arg2}$.
3. *[* 火车 晚点 *]*$_{arg1}$ 。 *[* 我 会在 九点 钟 到 那里 *]*$_{arg2}$ 。
   (*[The train is late]*$_{arg1}$. *[I will be there at nine o'clock]*$_{arg2}$.)

Consider how drastically the above discourse relations and their meaning changed, when we dropped the underlined discourse connectives. With some effort we can infer a new missing connective and with it a new meaning or sense of each non-`Explicit` relation. Examples 1. and 3. now represent a cause and its result (as if *so* is present), while example 2. lists personal preferences in a conjunction (as if *and* is present). In a natural language text it is quite common that discourse relations are not signalled by a discourse connective. Such situations can be difficult even for humans, because the sense needs to be inferred through the semantic context, coherence of arguments, knowledge about

the world, or other means [14]. From a computational perspective, such situations are much more challenging and represent a bottleneck of entire systems.

Discourse parsing is the task of extracting discourse relations. It involves locating `conn` and `punc`, extracting `arg1` and `arg2`, and *sense classification*, i.e. determine which meaning or sense label can be inferred. It turns out that constructing automated discourse parsers is notoriously difficult. Especially because distinctions between sense labels require subtle semantic judgements, but these can not be easily captured using traditional NLP features. To improve on this, two conferences CoNLL 2015 and 2016 organized a Shared Task [15, 16] that focused on discourse parsing and sense classification of shallow discourse relations on English and Chinese languages, which have sufficiently large datasets available. In both years sense classification was implemented in 40 competing systems for English and 10 for Chinese. Around half of the methods used conventional machine learning techniques such as SVM, MaxEnt and CRF models that rely on thousands to millions of hand-engineered features constructed from word categories and positions [13], production and dependency rules [17], neighbouring words, syntactic parse trees and part-of-speech (POS) tags [18], and cross-argument similarity features based on word pairs [19]. These generally make weak predictors of sense labels and increase the complexity of the solutions, but nevertheless work pretty well for `Explicit` relations. The other half of the methods used various neural network models and relied on pre-trained word embeddings combined with previously mentioned hand-engineered features. On word embeddings of each argument they separately apply either a variant of summation pooling [20], a convolutional neural network (CNN) [19, 21], or a recurrent neural network (RNN) [22], followed by a feed-forward neural network (FFNN). Although these black-box solutions perform better for sense classification of non-`Explicit` relations, they still achieve pretty poor performance in F_1-scores.

In general it turns out that non-`Explicit` sense classification is still the most challenging problem of various applications. Existing systems for sense classification use a complex pipeline of substantially different models to handle specific types and situations of discourse relations. These models require preprocessing, hand-engineered features, external resources, and extensive fine-tuning for each language and set of sense labels.

Motivated by the way how children acquire a language we move away from the weaknesses and complexity of existing systems for sense classification. We attempt to design a language-independent method for the task of sense classification of shallow discourse relations.

1.2 Problem definition

In this dissertation we focus on the task of *sense classification* of shallow discourse relations as described in the CoNLL 2016 Shared Task [16] and attempt to approach it in a *language-independent* manner.

> *Definition 1.1: Sense classification* of shallow discourse relations:
> A *sense label* or semantic class describes the meaning as which a discourse relation can be interpreted (e.g. contrast, causation, conjunction). Due to differences between languages a set of 10 sense labels is defined for Chinese (for a complete list see Tab. 2.2) and 21 sense labels for English (for a complete list see Tab. 2.3).
> Given two discourse arguments (`arg1`, `arg2`), an optional connective (`conn`), and optional punctuation (`punc`), our task is to predict the *sense label* of the discourse relation these represent.

To better illustrate the whole process let us imagine an automated discourse parser that scans an English newspaper article and attempts to locate discourse relations within sentences and across multiple sentences. At a given moment it is processing the text seen below. First it locates the discourse connectives (`conn`), then extracts the two discourse arguments (`arg1`, `arg2`) bound to it, and skips punctuation (`punc`), because this is ignores in English. Systems typically also determine that it is an `Explicit` relation within the same sentence where `arg1` is before `arg2`. All extracted text spans and information is passed to one or more components of the discourse parser that perform sense classification.

- text: *"[But]$_{arg2}$ if [there are more buyers]$_{arg1}$, then [it may be important]$_{arg2}$."*
- `arg1`: there are more buyers
- `arg2`: But it may be important
- `conn`: if then
- information: English, `Explicit`, within same sentence, `arg1` is before `arg2`

The goal of sense classification is to determine which one of the 21 sense labels for English best describes the meaning we infer from this information. In the above example

conn signals that it is a conditional relation which corresponds with the interpretation that more buyers might influence some important decisions. The sense classification component therefore concludes that the sense label is Contingency.Condition.

More details on shallow discourse relations can be found in Section 2.1.

Definition 1.2: Language-independent approach to NLP tasks:
We consider an approach to be *independent of a language*, if it was not designed specifically for a given language and does not require any preprocessing, hand-engineered features, external resources, or extensive fine-tuning for each language. With other words, it is language-independent with respect to its inputs and architecture, and applicable as such to very different languages.

Motivated by the way how children acquire a language we move away from the weaknesses and complexity of existing systems for sense classification. We approach it from a drastically different and language-independent perspective (for comparison see Tab. 2.5). In our opinion such a method for sense classification needs to consist of a single model to handle all types and specific situations of discourse relations (no differences between `Explicit` and other relation types, within-sentence and multi-sentence situations, the order of arguments). It should not perform any preprocessing of its input text spans, nor require any hand-engineered features or external resources, not even pre-trained word embeddings.

In Chapter 4 we present how we accomplished all this in our FR system, a novel method for sense classification of shallow discourse relations based on focused RNNs layer. We successfully applied it with almost the same model hyper-parameters on two substantially different languages, English and Chinese (without having any knowledge of Chinese).

1.3 Scientific contributions

In the light of our motivation and the importance of sense classification for different applications, we present in this dissertation the following scientific contributions (see corresponding chapters for terminology):

1. *Multi-dimensional RNN-attention mechanism* (focused RNNs).

   We present focused recurrent neural networks (focused RNNs), a novel neural network layer with an attention mechanism for constructing sentence/argument embeddings. Its purpose is to transform arguments of discourse relations into several vector subspaces that encode different aspects of the input text spans. At the time when the focused RNNs layer was first conceived (early 2016), only single-attention mechanisms that aggregate with a weighted average existed. Up to our knowledge, our approach is the first to present two new concepts and still differs greatly from other attention mechanisms found in related work. First, it is the first attention mechanism using multi-head or multi-dimensional attention weights, instead of attending to only a single aspect at a time. Second, it is the first attention mechanism using RNNs for production of attention weights, instead of computing them as the inner product with a query vector. Third, by computing all attention weights in one pass, instead of recomputing them for different query vectors when focusing on different aspects. Fourth, by using RNNs for aggregation of argument embeddings, instead of a sum of the weighted vectors. Our focused RNNs layer consists of a filtering RNN followed by a multiplicative filtering/gating mechanism that enables downstream RNNs to focus on different aspects of the input sequence and project it into several embedding subspaces. These argument embeddings can later be used for different NLP tasks, such as sense classification.

   This contribution is covered in Chapter 3. Its concept was introduced in Weiss & Bajec [22] and more details published in Weiss & Bajec [23].

2. *Language-independent method for sense classification* (FR system).

   We present a novel method for sense classification of shallow discourse relations based on focused RNNs layer, hence the name FR system. Up to our knowledge, our method presents a unique approach to sense classification that differs from

existing methods in many ways. First, it is the first using only a single end-to-end trainable model, instead of a complex pipeline of substantially different models to handle specific types and situations of discourse relations (no differences between `Explicit` and other relation types, within-sentence and multi-sentence relations, the order of arguments). Second, it is the first language-independent approach that requires no hand-engineered features or external resources, not even pre-trained word embeddings. It requires only a training dataset to work, which makes it usable almost out-of-the-box on any language and set of sense labels. Third, it is the first method that can be applied at the word and character level inputs without any preprocessing. Forth, it provides a simple data augmentation technique to produce more samples, instead of training only on given samples. We evaluate our method using the official datasets and methodology of CoNLL 2016 Shared Task. It does not fall a lot behind state-of-the-art performance on English, the most researched and supported language, but it outperforms other systems without a focused RNNs layer by 8% on the Chinese dataset. We first analyse its overall performance in terms of F_1-score and Brier-score, then in more detail with per-sense results and confusion matrices for `Explicit` and non-`Explicit` relations, and perform a case study of errors on English. We also analyse its training and classification time complexity. To qualitatively assess the contribution of some design choices we also perform an ablation study.

This contribution is covered in Chapter 4 and evaluated in Chapter 5. Our older more complex two-model system [22] received the first award by a large margin on Chinese datasets at CoNLL 2016 Shared Task. We generalized upon it and published the FR system in Weiss & Bajec [23].

1.4 Dissertation overview

In this dissertation we approach the sense classification task from the perspective of how children acquire a language. Instead of depending on external NLP resources, incorporating hand-engineered features, complex processing pipelines, and linguistic expert knowledge, we approach it in a language-independent manner. Although the performance of our method is affected by huge differences between languages, it is still remarkable that it can be used almost out-of-the-box on any language, set of sense labels, or level

of input representation. We feel that such an approach is not only beneficial for sense classification of shallow discourse relations, but will inspire researchers to also adapt it for even more complex natural language understanding tasks.

We organize the rest of this dissertation in five main chapters. Chapter 2 describes shallow discourse relations, surveys existing systems, and deep neural networks. Chapter 3 introduces our focused RNNs layer for constructing sentence/argument embeddings. Chapter 4 describes our FR system, a novel method for sense classification, its neural network layers, and implementation details. Chapter 5 continues with the evaluation and detailed analysis of its performance on Chinese and English languages. Chapter 6 makes a brief overview of achieved contributions and draws future research directions.

2

Background

In this chapter we provide an overview of shallow discourse relations and datasets used in CoNLL 2016 Shared Task. We continue with a comparison of existing systems for sense classification and typical modelling approaches. We finish with a brief introduction to deep neural networks on which our approach is based upon.

2.1 Shallow discourse relations

The goal of text-level discourse parsing is to discover the latent relational structure of a given well-written text or monologue, such as newspaper articles. Despite many decades of research, linguists were unable to discover clean theories of discourse grammars that can fully explain or describe how sentences are pieced together to form a coherent body of text, the same way we do in syntax. There are a few theoretical discourse frameworks along with annotated corpora [2, 11] that illustrate the theories and allow for computational investigation of discourse. Unfortunately, differences in theories, data set creation, features used, sets of sense labels, and experimental methodologies make it difficult to compare early works fairly and adequately. Nevertheless, it turned out that constructing text-level discourse parsers is notoriously difficult.

Different discourse frameworks differ in many ways, but they all more of less distinguish two problems that need to be solved by an automated discourse parser:

- *argument extraction*: determine the location and extent of text spans (e.g. clauses, phrases, sentences) that form a discourse relation
- *sense classification*: determine the meaning or sense label as which a discourse relation can be interpreted (e.g. contrast, causation, conjunction)

In this dissertation we have chosen the text-level discourse framework of *shallow discourse relations*, also called PDTB-style, because it strives to maintain a theory-neutral approach and offers the largest annotated corpora [12]. Each discourse relation is lexically anchored to a discourse connective (`conn`), even when it is not explicitly expressed, and takes two discourse arguments (`arg1`, `arg2`) as predicates. `conn` can be any of coordinating conjunctions (e.g. *and*, *or*), subordinating conjunctions (e.g. *because*, *when*, *since*), and discourse adverbials (e.g. *however*, *previously*, *nevertheless*). `arg1` and `arg2` may be clauses, noun phrases, sentences, and other non-contiguous text spans determined by the minimality principle that selects all and only the material needed to interpret

the discourse relation. Punctuation (punc) helps to determine the discourse relation in Chinese, but is otherwise ignored in English. Discourse relations are treated locally and independently, i.e. they are not connected to one another to form a global data structure, like a tree, and thus often overlap. They occur both across sentences and within sentences, the order of arg1 and arg2 is determined by the location of conn, and there are no restrictions on how many clauses and gaps they may contain. A *sense label* or semantic class describes the meaning as which a discourse relation can be interpreted (e.g. contrast, causation, conjunction). Due to differences between languages a set of 10 sense labels is defined for Chinese (for a complete list see Tab. 2.2) and 21 sense labels for English (for a complete list see Tab. 2.3).

The literature concerning shallow discourse relations further distinguishes the following four *relation types*, but focuses mostly on Explicit and Implicit relations that occur most often (see Tab. 2.1):

- Explicit relations use discourse connectives or markers as linguistic expressions that explicitly signal the presence of a discourse relation [24].

  "[Most oil companies]$_{arg1}$, when [they set exploration and production budgets for this year]$_{arg2}$, [forecast revenue of $15 for each barrel of crude produced]$_{arg1}$."

  — Explicit, sense Temporal.Synchrony

- For Implicit relations the discourse connective can be intuitively expressed, but is not lexically realized [12].

  "[Some have raised their cash positions to record levels]$_{arg1}$. [High cash positions help buffer a fund when the market falls]$_{arg2}$."

  — Implicit, sense Contingency.Cause.Reason

- Less frequent AltLex relations are realized by some alternative non-connective expressions and inserting a connective would lead to redundancy.

  "[Earnings fell to $877 million, or $1.51 a share]$_{arg1}$. [That compared with the year-earlier $1.25 billion, or $2.10 a share]$_{arg2}$ – which was inflated by the sale."

  — AltLex, sense Comparison.Contrast

- Less frequent EntRel entity-based coherence relations connect arguments only by the fact that they are about the same entity.

"[Hale Milgrim, 41 years old, was named president of Capitol Records Inc.]$_{arg1}$ [Mr. Milgrim succeeds David Berman, who resigned last month]$_{arg2}$."

— `EntRel`, sense EntRel

- Cases when no discourse relation can be perceived are marked with `NoRel`.

Table 2.1

Distribution by relation type in Chinese and English datasets. Each relation type further consists of sense labels, see Tab. 2.2 and Tab. 2.3.

	Chinese datasets				English datasets			
Relation type	train	valid	test	blind	train	valid	test	blind
`Explicit`	2225	77	96	566	14722	680	923	556
`Implicit`	6706	251	281	1399	13156	522	769	425
`AltLex`	211	5	7	49	524	19	30	28
`EntRel`	1098	50	71	87	4133	215	217	200
Total relations	10240	383	455	2101	32535	1436	1939	1209

Conferences CoNLL 2015 and 2016 organized a Shared Task [15, 16] that focused on *discourse parsing* and *sense classification* of shallow discourse relations on English and Chinese languages. Datasets are based on two corpora of newspaper articles, the English Penn Discourse TreeBank 2.0 (PDTB) [12] and the recently published Chinese Discourse Treebank 0.5 (CDTB) [25]. All datasets from the CoNLL 2016 Shared Task (http://www.cs.brandeis.edu/~clp/conll16st/) can otherwise be obtained from the Linguistic Data Consortium (LDC) repository (catalogue number LDC2017T13, http://catalog.ldc.upenn.edu/LDC2017T13). To make the task more approachable some sense labels with too subtle differences were collapsed, a dedicated sense label `EntRel` was assigned to entity-based coherence relations, `NoRel` relations were excluded, and any other supplementary information removed. Multiple sense labels are provided when the annotators have identified more than one simultaneous interpretation. As can be seen from Tab. 2.1, `Implicit` relations occur in English almost as often as `Explicit` ones, but in Chinese almost three times as often. Fig. 2.1 represents the distribution for the length of `arg1` and `arg2` measured as the number of words or number of characters. Length of both arguments is equally distributed and extremely long arguments are rare. Due to practical reasons (memory consumption, large training times) we can safely trucate arguments that are too long and know that less then 1% of samples are affected. CoNLL 2016 Shared Task defines the F_1-score as the most suitable measure for comparing the methods of the competition (we define it in Section 5.1).

In the competition it turned out that the most challenging problem is sense classification of `Implicit` relations. In our research we make a step back and focus on the whole task of *sense classification* of shallow discourse relations as described in the CoNLL 2016 Shared Task [16].

Fundamental differences between Chinese and English languages, such as the formalization of the concept of a sentence and the way arguments are labelled, also affect discourse relations. There are 10 sense labels defined for Chinese and 21 for English (for a complete list see Tab. 2.2 and Tab. 2.3 or CoNLL 2016 Shared Task [16]). These sense labels are unevenly distributed, especially in Chinese where more than half of relations signal the sense Conjunction or in English where almost a quarter of relations signal the sense Expansion.Conjunction. Because we are working on a language-independent approach, we primarily focus on Chinese datasets where less linguistic research and resources exists and benefits would be greater.

2.1.1 Chinese datasets

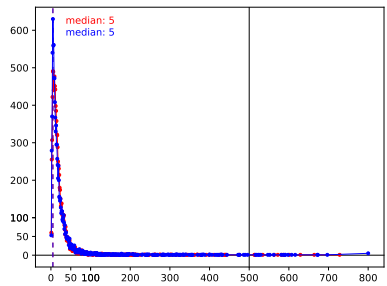
The Chinese datasets for the CoNLL 2016 Shared Task were adapted from the Chinese Discourse Treebank 0.5 (CDTB) [25] and the Chinese Wikinews [16]. The CDTB uses newswire articles from Xinhua News Agency and follows the general annotation strategy of the PDTB, but adapts it to the Chinese language. The train dataset contains 10240 relations from CDTB, the valid or development dataset contains 383, and the test dataset 455. The official ranking is based on the slightly out-of-domain blind test dataset to evaluate robustness, which contains 2101 relations from 64 articles from the Chinese Wikinews. The official datasets also provide additional layers of automatic linguistic annotation processed with state-of-the-art NLP tools (POS tags, syntactic parse trees, and dependency parses), but in our approach we ignore this information.

Since the concept of a sentence is less formalized in Chinese and punctuation plays an important role at disambiguating discourse relations, the discourse annotation scheme had to be adopted for Chinese. With the generalized lexically grounded approach there is no fundamental difference between `Explicit` and `Implicit` relations. Both arguments and sense labels need to be defined semantically, meaning they are defined based on how the arguments are interpreted. These allows the sense structure to be more general and less dependent on discourse connectives.

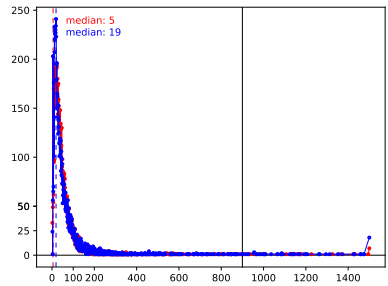
Consequently, we predict a flat set of 10 sense labels. Tab. 2.2 shows the number of occurrences of discourse relations according to sense labels and relation types. In

Figure 2.1

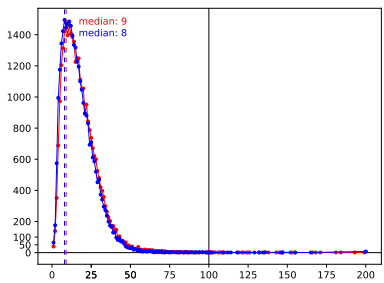
Word count (a)(c) and character count (b)(d) distributions for `arg1` (red) and `arg2` (blue) on Chinese and English train datasets.



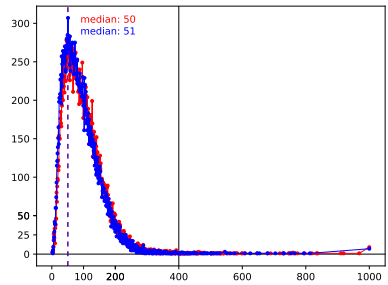
(a) Word count distribution on Chinese. Only 0.33% of relations have any argument longer than 500.



(b) Character count distribution on Chinese. Only 0.89% of relations have any argument longer than 900.



(c) Word count distribution on English. Only 0.19% of relations have any argument longer than 100.



(d) Character count distribution on English. Only 0.48% of relations have any argument longer than 400.

Chinese, `Implicit` relations occur almost three times as often as `Explicit` ones, while other types are less common. There are huge differences between relation types in the distribution of sense labels. Sense labels are much less evenly distributed in Chinese than in English, with sense CONJUNCTION occurring 59% of the time. The ground truth for official datasets was prepared by several expert human annotators. Their reported inter-annotator agreement score for sense labels on the train dataset is 87.4% with higher agreement for `Explicit` than for non-`Explicit` types of relations.

In the following Chinese example (also translated to English) we demonstrate how a segment of text can contain two overlapping discourse relations, and how conn can consist of multiple pieces:

Table 2.2 Distribution by sense labels and relation types in the Chinese train dataset.

Sense label	All	Explicit	Implicit	AltLex	EntRel
Alternative	18	18 (1%)	– (0%)	– (0%)	–
Causation	456	190 (9%)	187 (3%)	**79** (37%)	–
Conditional	134	100 (4%)	26 (0%)	8 (4%)	–
Conjunction	6102	**904** (41%)	**5174** (77%)	24 (11%)	–
Contrast	340	266 (12%)	66 (1%)	8 (4%)	–
EntRel	1098	– (0%)	– (0%)	– (0%)	**1098**
Expansion	1435	205 (9%)	**1188** (18%)	**42** (20%)	–
Progression	75	61 (3%)	7 (0%)	7 (3%)	–
Purpose	244	164 (7%)	56 (1%)	24 (11%)	–
Temporal	434	**383** (17%)	26 (0%)	25 (12%)	–
Total sense labels	10336	2291	6730	217	1098
Total relations	10240	2225	6706	211	1098

Sense labels that occur more than 15% of the time for each relation type are shown in bold. EntRel has a dedicated sense label. Less than 1% of discourse relations have multiple sense labels.

1. [建筑 公司 进 区]$_{arg1}$ ， [有关 部门 先 送上 这些 法规性 文件 ，然后 有 专门 队伍 进行 监督 检查]$_{arg2}$ 。
   ([Construction companies enter the area]$_{arg1}$, [relevant departments first send these regulatory documents, and then a special team conducts supervision and inspection]$_{arg2}$.)
   — Implicit, sense Conditional

2. 建筑 公司 进 区 ， [有关 部门 先 送上 这些 法规性 文件]$_{arg1}$ ，然后 [有 专门 队伍 进行 监督 检查]$_{arg2}$ 。
   (Construction companies enter the area, [relevant departments first send these regulatory documents]$_{arg1}$, and then [a special team conducts supervision and inspection]$_{arg2}$.)
   — Explicit, sense Temporal

2.1.2 *English datasets*

The English datasets for the CoNLL 2016 Shared Task were adapted from the Penn Discourse TreeBank 2.0 (PDTB) [12] and the English Wikinews [15]. The PDTB annotates discourse relations over the one million word corpus from Wall Street Journal. The train dataset contains 32535 relations from Sections 2-21 of the PDTB, the valid or develop-

ment dataset contains 1436 relations from Section 22, and the test dataset contains 1939 relations from Section 23. The official ranking is based on the slightly out-of-domain blind test dataset to evaluate robustness, which contains 1209 relations from 71 articles from the English Wikinews. The official datasets also provide additional layers of automatic linguistic annotation processed with state-of-the-art NLP tools (POS tags, syntactic parse trees, and dependency parses), but in our approach we ignore this information.

Sense labels in all English datasets are organized in a three-level hierarchy adopted from the PDTB. They even distinguish the semantic contribution or role of each argument. Discourse relations are always anchored to discourse connectives, even when they are not lexically realized. To reduce some sparsity without losing too much of the semantics, some sense labels from the original PDTB annotation have been merged and the attribute annotation from PDTB was removed.

As a result there are only 21 different sense labels, 15 completely and 6 partially annotated sense labels, that we need to predict. Tab. 2.3 shows the distribution of discourse relations according to sense labels and relation types. In English, `Implicit` relations occur in English almost as often as `Explicit` ones, while other types are far less common. There are huge differences between relation types in the distribution of sense labels. Sense labels are unevenly distributed, such that three most frequent sense labels (Expansion.Conjunction, Comparison.Contrast, and EntRel) occur 51% of the time. The ground truth for official datasets was prepared by several expert human annotators. Their reported inter-annotator agreement score for sense labels on the blind dataset is 85.5% with higher agreement for `Explicit` than for non-`Explicit` types of relations.

The following English example demonstrates how a segment of text can contain three overlapping discourse relations, and how `arg2` and `conn` can consist of multiple pieces:

1. *"[Kemper is using program trading]$_{arg1}$." He added [that "having one such firm doesn't matter]$_{arg2}$. But if there are more, then it may be important."*
   — `Implicit`, sense Comparison.Contrast

2. *"Kemper is using program trading." He added [that "having one such firm doesn't matter]$_{arg1}$. But [if there are more, then it may be important]$_{arg2}$."*
   — `Explicit`, sense Comparison.Concession

Table 2.3
Distribution by sense labels and relation types in English train dataset.

Sense label	All	Explicit	Implicit	AltLex	EntRel
Comparison	496	351 (2%)	145 (1%)	– (0%)	–
Comparison.Concession	1293	1093 (7%)	196 (1%)	4 (1%)	–
Comparison.Contrast	4714	**3024** (21%)	1657 (13%)	33 (6%)	–
Contingency	8	5 (0%)	3 (0%)	– (0%)	–
Contingency.Cause	1	– (0%)	1 (0%)	– (0%)	–
Contingency.Cause.Reason	3344	1168 (8%)	**2098** (16%)	78 (15%)	–
Contingency.Cause.Result	2137	601 (4%)	1389 (11%)	**147** (28%)	–
Contingency.Condition	1197	1193 (8%)	2 (0%)	2 (0%)	–
EntRel	4133	– (0%)	– (0%)	– (0%)	**4133**
Expansion	105	26 (0%)	77 (1%)	2 (0%)	–
Expansion.Alt	210	198 (1%)	12 (0%)	– (0%)	–
Expansion.Alt.Chosen alt.	241	99 (1%)	142 (1%)	– (0%)	–
Expansion.Conjunction	7817	**4414** (30%)	**3308** (25%)	**95** (18%)	–
Expansion.Exception	15	13 (0%)	1 (0%)	1 (0%)	–
Expansion.Instantiation	1403	236 (2%)	1134 (9%)	33 (6%)	–
Expansion.Restatement	2699	126 (1%)	**2514** (19%)	59 (11%)	–
Temporal	9	8 (0%)	1 (0%)	– (0%)	–
Temporal.Async	3	3 (0%)	– (0%)	– (0%)	–
Temporal.Async.Precedence	1277	801 (5%)	433 (3%)	43 (8%)	–
Temporal.Async.Succession	1014	870 (6%)	125 (1%)	19 (4%)	–
Temporal.Synchrony	1499	1271 (9%)	212 (2%)	16 (3%)	–
Total sense labels	33615	15500	13450	532	4133
Total relations	32535	14722	13156	524	4133

Sense labels that occur more than 15% of the time for each relation type are shown in bold. EntRel has a dedicated sense label. Around 3% of discourse relations have multiple sense labels.

3. *"Kemper is using program trading." He added that "having one such firm doesn't matter. [But]*$_{arg2}$ *if [there are more]*$_{arg1}$*, then [it may be important]*$_{arg2}$*."*
— Explicit, sense Contingency.Condition

2.2 Existing systems

The first complete *discourse parser* for shallow discourse relations was developed by Lin et al. [10] and it introduces a typical system architecture, like in Fig. 2.2, which was pursued and upgraded by subsequent work. In this architecture the parser works in a complex multi-step pipeline and consists of multiple components, such as discourse connective detector, argument extractor for `Explicit` relations, argument extractor for `Implicit` relations, argument re-ordering, discourse connective classifier, and sense classifier for `Explicit` and `Implicit` relations. These components use a fixed connective list to identify candidates, shallow features, POS tags, and parse trees to first extract and classify `Explicit` relations, followed by production and dependency rules and word-pair features to extract and classify non-`Explicit` relations. The end output of a complete system is precisely a list of discourse relations as they are annotated in datasets. Because of the pipeline-like architecture errors propagate to later processing steps and affect the performance.

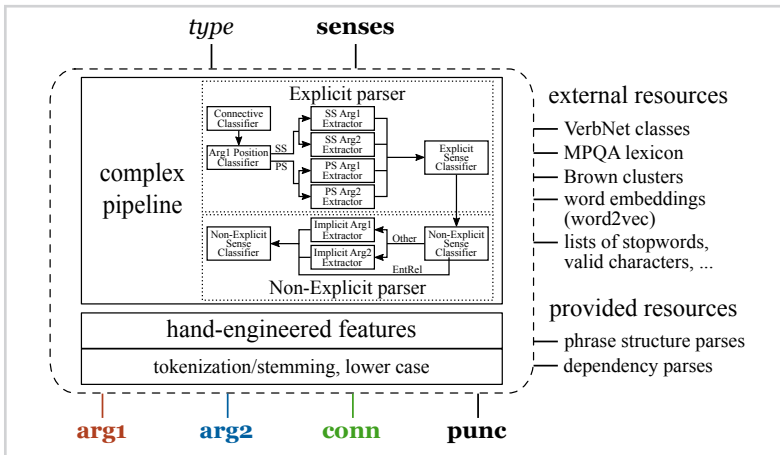


Figure 2.2
Typical system architecture of a shallow discourse parser with sense classification.

Best known systems for English adopt some variation of the typical pipeline architecture, usually adding even more hand-engineered features and production rules, heuristic argument extractors, ensemble of sense classifiers, and separately handling within-sentence and cross-sentence situations [18, 26]. Best known systems for Chinese modify the typical pipeline architecture similarly, but also use punctuation marks, a combina-

tion of classifiers and rules to determine the argument labels, and a seed-expansion approach to extract them [21, 27]. The performance of best known systems is presented in Tab. 2.4.

Table 2.4
Performance in F_1-scores of individual steps of the best discourse parsers on Chinese and English blind dataset.

	Chinese parser [21]			English parser [26]		
	All	Exp	Non-E	All	Exp	Non-E
(A) Only locating conn	–	0.6307	–	–	0.9179	–
(B) Only arg1/arg2 extr. (partial)	0.6839	0.5102	0.6238	0.8053	0.7174	0.8631
(C) Only arg1/arg2 extraction	0.4199	0.3181	0.4236	0.4824	0.4395	0.5202
(D) Only sense classification	0.6473	0.7669	0.6052	0.5354	0.7681	0.3366
(E) Overall parser	0.2660	0.2888	0.2474	0.2777	0.3445	0.2189

In a discourse parser these steps are usually interconnected and performed by components in a complex pipeline. If we would formulate them as tasks: (A) We are given a segment of text and need to mark all discourse connectives with conn. (B) We are given a segment of text and the marked conn and need to mark at least the first word of each argument with arg1 or arg2. (C) We are given a segment of text and the marked conn and need to mark the exact location and extent of each argument with arg1 or arg2. (D) We are given the text spans marked with arg1, arg2, conn, and punc and need to predict the sense label. (E) We are given a segment of text and need to extract all text spans and predict the sense label.

Sense classifiers are either stand-alone systems or components of a discourse parser that perform sense classification. Tab. 2.5 presents a comparison of best performing methods for sense classification. All previous systems or components for sense classification consist of substantially different models for English and Chinese languages, but also for handling Explicit, Implicit and other relations types. Most of them are highly language-dependent and require many external resources. In our FR system, either at the word level (FR-zh, FR-en) or at the character level (FR-zhch, FR-ench), we pursue a language-independent approach that differs in many ways, as illustrated in Fig. 2.3.

2.2.1 *Sense classifiers for Explicit relations*

Explicit relations use discourse connectives (conn) or discourse markers as linguistic expressions that explicitly signal the presence of a discourse relation between two discourse arguments (arg1, arg2) [24]. It turns out, that using only a list of connectives sets a reasonably high baseline for sense classification on English. Adding more syntactic category features helps to mitigate most ambiguities between their discourse or non-discourse usage [13]. Further improvements can be achieved by also extracting POS tags and features from the context of connectives [18].

Table 2.5

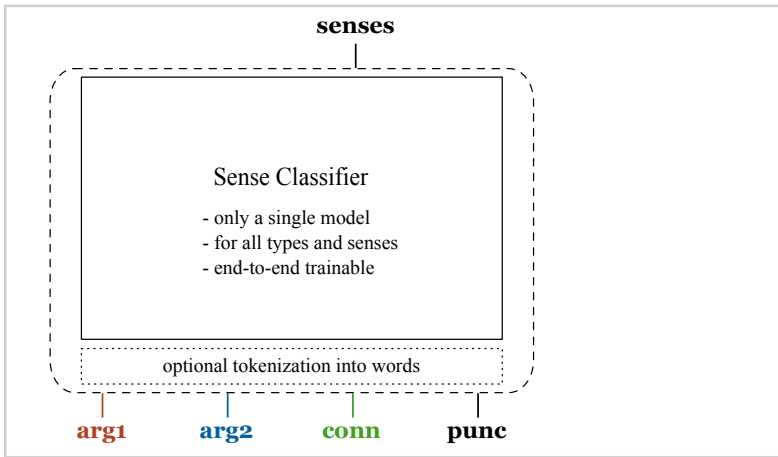
Comparison of best systems or components for sense classification for shallow discourse relations.

	[26]	[19]	[20]	[21]	[28]	FR system [23]
supported languages	en	en	en, zh	en, zh	en, zh	en, zh
supported relation types	`All`	`All`	`Non-E`	`All`	`All`	`All`
different sense models	6	2	1	5	3	1
external resources	3	2	1	3	2	0
hand-engineered features	yes	yes	no	yes	no	no
end-to-end trainable	no	no	yes	no	no	yes
tokenized input	required	required	required	required	required	optional

[26] is the best discourse parser for English, [19] the overall best sense classifier for English, [20] the best non-`Explicit` sense classifier for English, [21] the best discourse parser for Chinese (with sense classification results), [28] the second best `Explicit` sense classifier for Chinese, and our FR system [23] the overall best sense classifier for Chinese.

Figure 2.3

Simplified system architecture of a language-independent approach for sense classification.



Best `Explicit` sense classifiers: For English, the best known `Explicit` sense classifier [19] uses a logistic regression classifier with several cross-argument similarity features based on pre-trained word embeddings. For Chinese, our FR-zhch system [23] (previously called FR-ca) with focused RNNs layer works best, while the second best uses a SVM on the connectives themselves [28]. Other methods for Chinese `Explicit` relations also use a logistic regression classifier and features similar to English [21], but their performance is slightly worse.

2.2.2 *Sense classifiers for* `Implicit` *relations*

`Implicit` relations are missing an explicit discourse connective and their interpretation needs to rely on the meaning and general knowledge about the world. Half of the approaches use *conventional machine learning* techniques and rely on hand-engineering. Due to the lack of data, early work used patterns to extract `Explicit` relations from unlabelled data, and generated examples of synthetic `Implicit` relations by just removing the connectives [29]. However, Rutherford & Xue [30] later showed that linguistic dissimilarity between `Explicit` and `Implicit` relations has to be considered to determine in which cases it is safe to do this. Early supervised approaches relied heavily on lexicons and hand-engineered features derived from syntactic parse trees. Attempts at using features based on cross-products of words between arguments helped, but they are not the semantically-related pairs that researchers hoped for [14]. Another attempt that somewhat helped was to produce millions of features from production rules, dependency rules, and word pairs followed by a feature selection process [17, 31]. Employing additional millions of binary features from Brown cluster pairs and coreference patterns improves the results even further [18, 32]. The crucial role of feature selection and cut-off thresholds indicates that most features are useless and contribute more noise than signal.

Best `Implicit` *sense classifiers with conventional machine learning*: For English, Naive Bayes classifier with millions of features and feature selection proved to be the most efficient and consistently best-performing conventional machine learning technique for `Implicit` relations [31, 32]. For Chinese, the best conventional methods are based on production rules of arguments [27], however some managed to enhance them with pairs of words and verbs at specific locations to achieve slightly better performance [21].

On the other hand, methods based on *neural networks* seem to be particularly appealing for processing `Implicit` relations due to their power of capturing semantic information in high-dimensional dense vector representations. The general approach is to first construct vector representations of both discourse arguments, called argument embedding, and use them to perform sense classification. An early approach constructed argument embeddings as an average of pre-trained word embeddings combined with Brown clusters [33]. Another approach computed vector representations of discourse arguments and coreferent entity mentions through a series of compositional operations over the syntactic parse tree [34]. Braud & Denis [35] performed a comparison of one-

hot representations, Brown clusters, and pre-trained vector representations on word pairs from discourse arguments. It showed that pre-trained word embeddings seem to provide most of the semantic and syntactic information relevant for the task.

Best `Implicit` sense classifiers with neural networks: Best known approaches with neural networks somehow construct argument embeddings and then apply a FFNN for classification. For English, the simplest approach computes only an average of pre-trained word embeddings and achieves state-of-the-art performance [20]. Some apply CNNs on each argument separately [19, 21], while others use them to produce shared vector representations of cross-argument word pairs in a multi-task environment with different annotation frameworks [36]. Another approach performs a series of summations and multiplications of pre-trained word embeddings and embeddings from the parse tree [28]. For Chinese, our FR-zh system [23] (previously called FR-wa) with focused RNNs layer works best. The previous state-of-the-art performance was achieved by our older and more complex two-model system with focused RNNs at word level [22]. It differs from the FR system [23] by: using two separate models (one for processing `Explicit` and one for non-`Explicit` relations), requires tokenized/segmented input at the word level, during training only random noise samples are introduced for each discourse relation sample, and it uses many fine-tuned parameters for each model and language. We build upon this approach, because we want the FR system to be easily trainable, handle different languages and sets of sense labels, and not depend on external resources.

2.3 Deep neural networks

Artificial neural networks can be viewed as machine learning technique or universal function approximator mapping input vectors of real numbers into output vectors that is loosely modelled after biological brains. It is composed of simple elements called artificial neurons, which receive input as a weighted sum from other neurons and produce their output using a non-linear activation function. By composing artificial neurons into different neural network architectures, we influence how information is being processed and which types of tasks they are suitable for. In the process of supervised learning from labelled data, the backpropagation algorithm is used to find the right weights that correctly transform the input into the labelled output.

Deep neural networks or deep learning is the name we use for stacked artificial neural

networks with many layers (i.e. networks composed of four or more layers). Each layer learns to perform feature extraction and transforms its input data into a slightly more abstract and composite representation. In end-to-end training setup it is even possible that the input to the first layer is just raw vectorized high-dimensional data and the output of the last layer the complete low-dimensional solution of a given task. They perform automatic feature extraction without human intervention, unlike most conventional machine learning techniques. The extra layers enable a hierarchical composition of features from lower layers, potentially modelling complex data with fewer units than a similarly performing shallow artificial neural network (i.e. network with three or less layers).

Simplest and most common *feed-forward neural networks* (FFNN) are inappropriate for most NLP tasks, because text represents a sequence of words with variable length. *Recurrent neural networks (RNNs)* [37] or their generalization, recursive neural networks [38], solve this by applying the same set of weights over a sequence (temporal dimension) or a structure (tree-based). A commonly used RNN is the Long short-term memory (LSTM) layer [39] that is capable of storing information in a memory cell over extended time intervals. Theoretically, this allows it to extract semantic information from words, accumulate it, and produce a semantic representation of the whole sentence on the end. A bidirectional LSTM layer [40] applies one LSTM in the forward and one in the backward direction, and then combines both outputs. It can therefore incorporate information from preceding as well as following words in a sentence. Recent success of deep neural networks in NLP tasks was made possible with breakthroughs in new neural network architectures (like embeddings, GRU, tensor networks, encoder-decoder frameworks, attention mechanisms, self-attention), training techniques (like RMSProp or Adam optimization, dropout regularization, batch normalization), initialization using unsupervised pre-training on massive datasets (like Skip-gram from Word2vec, GloVe, FastText), and faster computing resources (parallelization on GPUs).

Overall, deep neural networks seem like a suitable approach for accomplishing our sense classification task in a language-independent manner without any hand-engineered features or external resources, just like children do.

3

Multi-dimensional RNN-attention mechanism (focused RNNs)

In this chapter we first describe the related work on neural embeddings and attention mechanisms. Then we present the details of our focused RNNs layer for constructing sentence/argument embeddings.

3.1 Sentence/argument embeddings

An *embedding* is a mapping from discrete objects with no natural vector representation, such as words or sentences, into dense vectors of real numbers. It is known that neural networks train best on dense vectors, where individual dimensions typically have no inherent meaning and all values as a whole contribute to define an object. Ideally, an embedding captures some of the semantics of the input by placing semantically similar inputs close together in the vector space. Embeddings are an important and effective way for transforming the input for neural networks.

Over the last five years many possible ways of constructing *word embeddings* or word representations in a vector space have been proposed. The most commonly used models are Skip-gram from Word2vec [41] and GloVe [42] which are both unsupervised approaches based on the distributional hypothesis (words that occur in the same contexts tend to have similar meanings). Most NLP methods with neural networks depend on these pre-trained word embeddings trained on huge datasets. Only a few, including our FR system, train their own task-specific word embeddings.

For NLP classification tasks a typical approach with neural networks is to first map words into word embeddings, then encode sentences or pieces of text into fixed-length task-specific vector representations, also called *sentence/argument embeddings*, and then apply a FFNN for classification. A variation of this basic structure is also followed by all neural network models for sense classification of `Implicit` relations, including our FR system (see Chapter 4). The most distinguishing step is how sentence embeddings are produced. In our case they are called argument embeddings and it is crucial how much of the semantic similarity and coherence information related to discourse relations they capture. For many NLP tasks it has been shown that a simple approach of just computing the average vector of all word embeddings in a sentence, called Bag-of-Words approach, gives a strong baseline. The Bag-of-Words approach loses all information on words ordering and local features, but preserves a surprising amount of semantic and syntactic content of discourse relations [20]. A more advanced approach is to use CNNs to extract local features and perform max-pooling over time. Such an approach still treats all

features as equally important and loses information about their order [19, 21]. A typical alternative is to use one or more layers of bidirectional LSTMs that can theoretically learn a complex aggregation function and preserve all required local information in a sentence embeddings (see simple LSTMs baseline in Chapter 5). Unfortunately, because of the overwhelming complexity of natural language text such approaches require a high-dimensional sentence embedding representation, huge datasets, and extremely long training times.

Neural attention mechanisms explore the fact that all words do not equally contribute to the meaning of a sentence and different parts contain different knowledge. In general, attention mechanisms allow a model to automatically search for the parts of the input that are relevant for processing at each step and adjust its focal point over time. They have become an integral part of models that must capture global dependencies or attend to different parts. In particular, given an sequence of word embeddings $x = [x^{(1)}, x^{(2)}, \dots, x^{(m)}]$ and the vector representation of a query vector q, the attention mechanism computes an alignment score or weight between $x^{(t)}$ and q by a comparison function $f^{(t)} = \mathrm{F}(x^{(t)}, q)$, which measures how important $x^{(t)}$ is to q on a specific task. These attention weights are then used to scale the input sequence $a^{(t)} = f^{(t)}x^{(t)}$ that is passed into an aggregation function $b = \mathrm{B}(a)$, which produces the sentence/argument embedding. Typically the output of an attention mechanism is just a weighted average of of all word embeddings in x.

Initially attention mechanisms were applied in encoder-decoder frameworks, such as image caption generation where a CNN encodes an image and the attention mechanism helps the RNN decode better textual descriptions [43]. For machine translation, the same concept, but with a bidirectional RNN for encoding, successfully learned to align words and translate between English and French [44]. Adding the attention mechanism to a three-layer LSTM model enabled it to successfully perform linearised syntactic constituency parsing [45]. Attention mechanism was also found to be suitable for question answering tasks [46], because it gives the model at each answer generating step a fuzzy access to its internal memory as a weighted average representation of all memory locations. End-to-end memory networks [47] present a different approach by stacking multiple attention layers and updating the question representation at each step. Since traditional attention computes a single attention weight for each word based on the word embedding, it cannot distinguish the meanings of the same word in different contexts. This is why after 2016 many approaches started using multi-head or multi-dimensional at-

tention weights to attend to multiple query vectors or aspects at once, each one best describing its meaning specific for a given aspect or context of a task. The concept of self-attention mechanisms [48], also called intra-attention, calculate the attention weight at a position in a sequence by attending to all positions within the same sequence. A sequence-to-sequence (seq2seq) model, called the Transformer [49], used an encoder-decoder structure that is only composed of stacked self-attention networks, without using either recurrence or convolution, and achieved state-of-the-art performance on the machine translation task. On the task of aspect-based sentiment analysis state-of-the-art performance was achieved by considering the relevance of sentiment words with respect to the given aspects [50]. They used a content attention mechanism to capture the important information about given aspects from a global perspective, and a context attention mechanism to take the order of the words into account.

Our concept of *focused RNNs layer* at word level was introduced by our older two-model system [22]. Up to our knowledge, our approach is the first to present two new concepts and still differs greatly from other attention mechanisms found in related work. In contrast to previous mechanisms, all attention weights are computed in one pass by a filtering RNN and not recomputed for individual query vectors when it needs to focus on different aspects. Instead of computing a single attention weight for each word, the multi-dimensional approach represents a natural progression of this idea by computing multiple attention weights for each aspect of each word in parallel. Instead of using a primitive weighted average to compute sentence/argument embeddings it applies a series of RNNs, called focused downstream RNNs, as an aggregation function to accumulate information on different aspects into a series of argument embedding subspaces. These vectors are concatenated into an argument embedding, so that it can be used for different NLP tasks, such as sense classification. Our generalization of focused RNNs layer [23] further improves upon this by: processing any sequence of input symbols of arbitrary lengths (such as character level inputs), sharing weights between multiple focused RNNs layers, using a bidirectional LSTM for filtering RNN, and LSTMs for focused downstream RNNs.

3.2 *Focused RNNs layer*

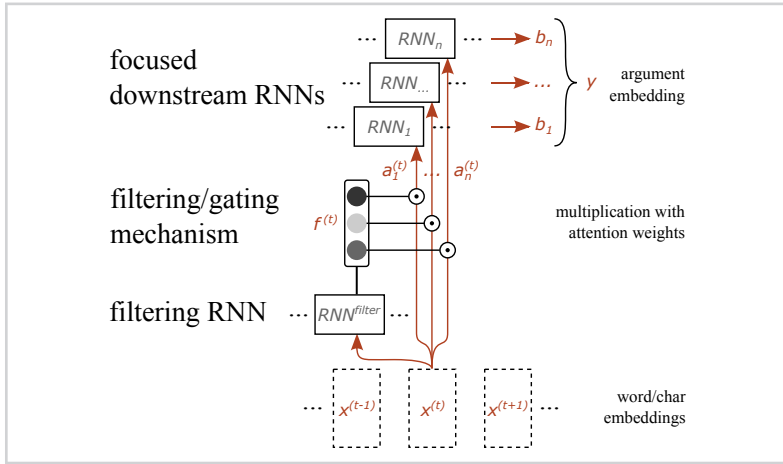
For creating better embeddings we constructed of a novel neural network layer called focused recurrent neural networks (*focused RNNs*), the first multi-dimensional RNN-attention mechanism for constructing sentence/argument embeddings. It is a general concept that consists of a *filtering RNN* (RNN^{filter}), a multiplicative *filtering/gating mechanism*, and *focused downstream RNNs* (RNN_i). We introduced it in Weiss & Bajec [22] and published more details in Weiss & Bajec [23].

Fig. 3.1 presents the processing diagram of our focused RNNs layer with n focused downstream RNNs. Its input can be any sequence of dense vectors we would like to transform into several vector subspaces. For sense classification, input is a sequence of word embeddings $x = [x^{(1)}, x^{(2)}, \dots, x^{(m)}]$ that represents a sentence or argument of a discourse relation of length m we want to encode into an argument embedding y. First the filtering RNN (RNN^{filter}) acts as an multi-dimensional comparison function and produces for each word embedding a vector of attention weights $f^{(t)}$. The filtering/gating mechanism multiplies each weight $f_i^{(t)}$ with the same word embedding to produce a weighted word embedding $a_i^{(t)}$ for one focused downstream RNN. The weighted input sequence produced in this way makes it possible for focused downstream RNNs (RNN_i) to specialize or focus on different aspects. Each focused downstream RNN acts as an aggregation function and accumulates in its internal state a fixed-size vector representation b_i. This represents a projection of the input sequence into an argument embedding subspace. Finally, all produced vectors b_i are concatenated/stacked into a longer vector y that represents the sentence/argument embedding and can be used for various NLP tasks, like sense classification.

The intuition behind it is that different downstream RNNs can specialize or focus on different aspects of each text span (e.g. discourse argument) in parallel and independently of one another. The filtering RNN directs their attention with a vector of attention weights (multi-dimensional attention mechanism) based on the input sequence itself (self-attention mechanism) and without explicit query vectors. In comparison to related work, the filtering RNN is also capable of learning a much more expressive comparison function than a dot-product with a query vector. Unfortunately, due to the black-box nature of neural networks it is unclear what these aspects represent. Note that the concept of focused RNNs layer differs greatly from other neural attention mechanisms. To the best of our knowledge, it represents the first multi-dimensional RNN-attention

Figure 3.1

Our focused RNNs layer architecture for constructing sentence/argument embeddings.



For each word embedding $x^{(t)}$ the filtering RNN produces a vector of attention weights $f^{(t)}$. The filtering mechanism multiplies each weight $f_i^{(t)}$ with the same word embedding to produce a weighted word embedding $a_i^{(t)}$. Focused downstream RNNs project these into several argument embedding subspaces b_i that are concatenated into an argument embedding y.

mechanism. Differences are described in Section 3.1.

Chosen hyper-parameters of our FR system in Tab. 4.1 indicate that the optimal number of focused downstream RNNs n depends more on the language, than the number of sense labels or their distribution. If the focused RNNs layer is used multiple times in a neural network, the same focused RNNs layer with the same trainable matrices should be applied to each text span. Specifically, the trainable weights of all filtering RNN should be shared globally to ensure that the attention weights for each text span are produced by the same mechanism. The same set of focused downstream RNNs (RNN_i) should be applied to each text span. This sharing encourages the i-th downstream RNN to specialize or focus on a different aspect and project it into the same argument embedding subspace, instead of overfitting on specifics of each text span. Experiments with our FR system have shown that disabling the sharing of trainable parameters degrades its performance.

3.2.1 *Filtering RNN*

First, for each position (t) of the input sequence the *filtering RNN* (RNN^{filter}) produces a vector of attention weights $f^{(t)}$, where $f_i^{(t)} \in [0, 1]$. An attention weight can be interpreted as the relative importance of how important $x^{(t)}$ is for a specific aspect or context of a task. Theoretically, any type of RNN can be used for the filtering RNN, but a bidirectional LSTM layer [40] with the σ activation function performs somewhat better. The LSTM layer [39] is a commonly used RNN that can aggregate and store information in the internal memory cell $c^{(t)}$ over extended time intervals. When combined in a bidirectional setting it can capture long-term dependencies from preceding and succeeding input symbols. The manipulation and usage of the internal memory cell $c^{(t)}$ is controlled with an input g^i, forget g^f, and output g^o gates. The LSTM layer is computed as

$$\begin{aligned} g_I^{(t)} &= \sigma(\mathrm{W}_I x^{(t)} + \mathrm{U}_I \vec{f}^{(t-1)}) \\ g_F^{(t)} &= \sigma(\mathrm{W}_F x^{(t)} + \mathrm{U}_F \vec{f}^{(t-1)}) \\ g_O^{(t)} &= \sigma(\mathrm{W}_O x^{(t)} + \mathrm{U}_O \vec{f}^{(t-1)}) \\ c^{(t)} &= g_F^{(t)} \odot c^{(t-1)} + g_I^{(t)} \odot \sigma(\mathrm{W}_C x^{(t)} + \mathrm{U}_C \vec{f}^{(t-1)}) \\ \vec{f}^{(t)} &= g_O^{(t)} \odot \sigma(c^{(t)}) \end{aligned} \tag{3.1}$$

where $\odot$ represents element-wise multiplication (Hadamard product). W_I, W_F, W_O, W_C, U_I, U_F, U_O, and U_C are trainable matrices, and $c^{(-1)}$ and $\vec{f}^{(-1)}$ are the initial hidden states. The bidirectional LSTM layer has two sets of formulas from Eq. 3.1 differing only in the direction of processing the time dimension. The final vector of attention weights $f^{(t)}$ is computed as an average of output vectors at matching positions as

$$f^{(t)} = \frac{\vec{f}^{(t)} + \overleftarrow{f}^{(t)}}{2} \tag{3.2}$$

where $\vec{f}^{(t)}$ and $\overleftarrow{f}^{(t)}$ represent output vectors from both directions. To give the LSTM layer more flexibility, the attention weights do not have to sum up to 1.

3.2.2 *Filtering/gating mechanism*

Afterwards we apply a multiplicative *filtering/gating mechanism* to regulate how much of the input signal should be passed to individual focused downstream RNNs. For each position (t) of the input sequence we have an input vector $x^{(t)}$, usually a word embedding, and a vector of attention weights $f^{(t)}$ from the filtering RNN. We multiply each

input vector $x^{(t)}$ with the scalar value of each dimension of the attention weights vector $f_i^{(t)}$ for all $i \in [1, n]$. The filtering/gating mechanism is computed as

$$\begin{aligned} f^{(t)} &= RNN^{filter}(x^{(t)}|x) \\ a_i^{(t)} &= f_i^{(t)} x^{(t)} \end{aligned} \tag{3.3}$$

where $RNN^{filter}(\cdot)$ is a function representing the filtering RNN, and $a_i^{(t)}$ the weighted vector to be passed to the i-th focused downstream RNN (RNN_i). With other words, one attention dimension scales the inputs of one downstream RNN to direct its attention to different aspects of the input sequence.

3.2.3 *Focused downstream RNNs*

The i-th *focused downstream RNN* (RNN_i) receives a sequence of weighted vectors $a_i = [a_i^{(1)}, a_i^{(2)}, \ldots, a_i^{(m)}]$ in order to project them into an argument embedding subspace b_i, where $i \in [1, n]$. Theoretically, any type of RNN can be used for focused downstream RNN and we did not observe any substantial gains in using specific RNNs. To avoid introducing new algorithms we also use an LSTM layer here with the same set of formulas from Eq. 3.1, this time only in the forward direction of processing. Each focused downstream RNN acts as an aggregation function and accumulates in its internal state a fixed-size vector representation of its aspect b_i. For usage in classification tasks, such as sense classification, all produced vectors b_i are concatenated/stacked into a longer vector y that represents the final sentence/argument embedding. The final sentence/argument embedding is computed as

$$\begin{aligned} b_i &= RNN_i(a_i^{(m)}|a) \\ y &= b_1 \| b_2 \| \ldots \| b_n \end{aligned} \tag{3.4}$$

where $RNN_i(\cdot)$ is a function representing the i-th focused downstream RNN, b_i is its last internal state when the whole sequence is processed, and operator $\|$ represents concatenation of vectors.

These sentence/argument embeddings can be used for various NLP tasks, such as sense classification.

4

Language-independent method for sense classification (FR system)

In this chapter we present our method for sense classification, called the FR system. We first present our approach, describe the details for each of the neural network layers it consists of, and finally describe the training procedure and implementation details.

4.1 Our approach

In this dissertation we focus on the two problems we defined in the Section 1.2. Providing a method for the task of *sense classification* of shallow discourse relations and attempting to approach it in a *language-independent* manner.

In Section 2.2, we review existing systems for sense classification. It turns out that all are build on a complex pipeline of substantially different models. These models require preprocessing, hand-engineered features, external resources, and extensive fine-tuning for each language and set of sense labels, but still perform somewhat poorly. Motivated by the way how children acquire a language we move away from the weaknesses and complexity of existing systems for sense classification. We approach it from a drastically different and language-independent perspective. In our opinion such a method for sense classification needs to consist of a single model to handle all types and specific situations of discourse relations (no differences between `Explicit` and other relation types, within-sentence and multi-sentence situations, the order of arguments). It should not perform any preprocessing of its input text spans, nor require any hand-engineered features or external resources, not even pre-trained word embeddings.

We accomplished all this in a novel method for sense classification of shallow discourse relations based on focused RNNs layer, hence the name FR system. It differs from existing systems in many ways as presented in Tab. 2.5. Because of its generic design it can be easily adapted to other NLP classification tasks for which we need end-to-end training and multi-dimensional argument embeddings. We successfully applied almost the same model hyper-parameters on two substantially different languages and two levels of input representation. This results in four different settings:

- *FR-zh system*: FR system for Chinese using word level representations
- *FR-zhch system*: FR system for Chinese using character level representations
- *FR-en system*: FR system for English using word level representations
- *FR-ench system*: FR system for English using character level representations

In Chapter 5, we analyse its performance on Chinese and English in terms of F_1-score and Brier-score, in more detail with per-sense results and confusion matrices for `Explicit` and non-`Explicit` relations, perform a case study of errors on English, analyse its training and classification time complexity, and perform an ablation study.

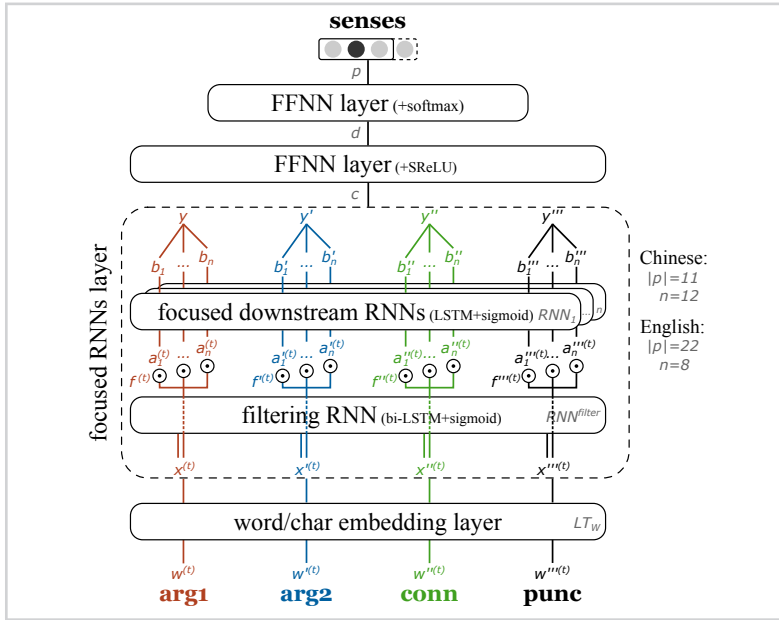
4.2 FR system

The *FR system* is our proposed method for sense classification of shallow discourse relations. It consists of a single end-to-end trainable neural network model, an input preparation step, and the training procedure with a simple data augmentation technique. Our model directly follows the sense classification task definition and handles all types and specific situations of discourse relations. It also requires no feature engineering or external resources, which makes it language-independent with respect to its inputs and architecture. It is also the first system for sense classification that can be applied at the word level (FR-zh, FR-en) and character level (FR-zhch, FR-ench) representation of input text spans. To improve on generalizability and project all text spans into the same argument embedding space, we apply the same set of layers to each text span, i.e. share their trainable parameters. Since we are learning task-specific word embeddings from scratch, we merely introduce a simple data augmentation technique during training. Because the model is end-to-end differentiable, it can be trained with the backpropagation algorithm on labelled samples of discourse relations. We published the FR system in Weiss & Bajec [23].

Fig. 4.1 presents the neural network architecture of our FR system for sense classification based on the focused RNNs layer (described in Chapter 3). Our method is given a discourse relation represented as raw input in the form of four text spans: for two arguments (`arg1`, `arg2`), an optional connective (`conn`), and optional punctuation (`punc`). In the spirit of end-to-end training we perform no preprocessing and work directly with the text spans represented as sequences of input symbols at either the word or character level ($w = [w^{(1)}, w^{(2)}, \ldots, w^{(m)}$). For consistency we process all text spans in exactly the same way, i.e. following the same equations. When necessary, we distinguish variables with apostrophes (e.g. w for `arg1`, w' for `arg2`, w'' for `conn`, and w''' for `punc`). Each text span is processed independently from others from its beginning till the end, where (t) is the current position in the time dimension (e.g. t-th word of a text span). First, the *word/char embedding layer* learns to transform input symbols into task-specific vec-

Figure 4.1

Our FR system neural network architecture for sense classification.



Each of the four text spans w is first mapped into word embeddings x, then independently processed by our focused RNNs layer to produce an argument embedding y. These are passed into a two-layer FFNN to predict the probabilities p for sense classification.

tor representations, called word or character embeddings $x^{(t)}$. Second, each sequence of word embeddings ($x = [x^{(1)}, x^{(2)}, \ldots, x^{(m)}]$) is then independently processed by our focused RNNs layer. The focused RNNs layer consists of a filtering RNN, a multiplicative filtering/gating mechanism, and several focused downstream RNNs. These project each sequence of word embeddings into a fixed-size vector representation, called argument embedding y. Then we concatenate argument embeddings of all text spans (y for `arg1`, y' for `arg2`, y'' for `conn`, and y''' for `punc`) into a longer vector c and pass it into a two-layer feed-forward neural network (FFNN). Its purpose is to predict the probabilities p of sense labels. Finally, FR system returns the sense label with the highest probability p as the result of sense classification for the given discourse relation.

We successfully applied the same neural network architecture on Chinese and English datasets at both the word level (FR-zh, FR-en) and character levels (FR-zhch, FR-ench).

We do not use any resources provided by the CoNLL 2016 Shared Task [16], such as POS tags, syntactic parse trees, dependency parses, Brown clusters, or pre-trained word embeddings. Due to the differences between languages and different sense labels, a few basic hyper-parameters had to be adjusted (see Tab. 4.1). All hyper-parameters and other design choices for each setting were fit by using the train and valid datasets for both languages. In Chapter 5 we present detailed evaluation results of each setting.

4.2.1 Word or character embedding layer

The first layer of our model transforms input symbols at either word or character level ($w^{(t)}$) into task-specific vector representations ($x^{(t)}$) suitable for neural networks, called word embeddings [51] or character embeddings.

FR system at the word level (FR-zh, FR-en): If we process the raw input at the word level, we use the fact that all datasets provide each discourse relation already in the form of four text spans already tokenized/segmented into words. We represent them as four sequences of words or tokens ($w^{(t)}$), in case a text span is non-contiguous we join all parts together. Due to practical reasons (i.e. memory consumption and large training times), we crop/truncate arguments that are too long in such a way that less then 1% of samples are affected (see the distribution of argument lengths in Fig. 2.1). In contrast to a typical NLP approach, we perform no stemming or lemmatization, conversion to lower case, accent stripping, Unicode normalization, removal of stopwords or invalid chars, or similar. To build a vocabulary of known words, which is needed for this layer to work, our method initially scans the training dataset. A special out-of-vocabulary symbol (`[OOV]`) is reserved for unseen words that may be encountered in other datasets.

FR system at the character level (FR-zhch, FR-ench): If we process input at the character level, we represent each discourse relation in the raw form of four text spans. We represent them as four sequences of characters ($w^{(t)}$), these include white-spaces, punctuation, and other symbols. There are far fewer different characters than there are words and each sentence contains more characters than words. Due to practical reasons (i.e. memory consumption and large training times), we crop/truncate arguments that are too long in such a way that less then 1% of samples are affected (see the distribution of argument lengths in Fig. 2.1). To build a vocabulary of known characters, which is needed for this layer to work, our method initially scans the training dataset. A special out-of-vocabulary symbol (`[OOV]`) is reserved for the very improbable event that unseen characters are encountered in other datasets. There are several benefits of using character-

level representations over word-level: do not require any tokenisation/segmentation, are is tolerant to textual errors, do not suffer from out-of-vocabulary issues, work in open-vocabulary situations, and are able to model different and rare morphological variants of a word. To avoid unnecessary long sentences, we will use the name word embeddings to also mean character embeddings when used in a general sense throughout this dissertation.

Both *word and character embedding layers* are computed in the same way. They can be represented as a lookup table $LT_{W_E}(\cdot)$ that maps each input symbol $w^{(t)}$ from the vocabulary into a fixed-size vector representation, called word or character embedding $x^{(t)}$,

$$\begin{aligned} x^{(t)} &= LT_{W_E}(w^{(t)}) \\ x'^{(t)} &= LT_{W_E}(w'^{(t)}) \\ x''^{(t)} &= LT_{W_E}(w''^{(t)}) \\ x'''^{(t)} &= LT_{W_E}(w'''^{(t)}) \end{aligned} \tag{4.1}$$

where (t) is the current position in a text span, $w^{(t)}$ the t-th word of a text span, and W_E a trainable matrix for the lookup table.

To better illustrate this process, let us consider the example in Section 1.2:

- text: *"[But]$_{arg2}$ if [there are more buyers]$_{arg1}$, then [it may be important]$_{arg2}$."*
- `arg2` at word level ($|w'| = 5$):
  $w' =$ ["But", "it", "may", "be", "important"]
- `arg2` at character level ($|w'| = 23$):
  $w' =$ ["B", "u", "t", " ", "i", "t", " ", "m", "a", "y", " ", "b", "e", " ", "i", "m", "p", ...]

We train these embeddings completely from scratch, but also performed experiments with pre-trained word embeddings as fixed values or used for initialization. Pre-trained word embedding lookup tables exist for different languages, such as Skip-gram from Word2vec [41] or GloVe [42], but experiments did not show any substantial improvements. On the other hand, there are no pre-trained character embeddings and they had to learn task-specific character embeddings from scratch. These embeddings automatically emerge when training the whole model in an end-to-end manner using the back-propagation algorithm. Even though some experiments suggest that optimal word embeddings are dependent on discourse relations [35], the lack of large amounts of training

data makes it unrealistic to learn separate word embeddings. We apply the same word or character embedding layer to all four text spans, i.e. share the trainable matrix W_E, and consequently all word embeddings $x^{(t)}$ are represented in the same vector space.

4.2.2 *Focused RNNs layer*

The *focused RNNs layer* can analyse different aspects of a sequence of word embeddings $x^{(t)}$ and project them into a fixed-size vector representation, called sentence or argument embedding y. Details are described in Chapter 3 and Weiss & Bajec [23].

In this layer, each text span is represented as a sequence of word embeddings $x = [x^{(1)}, x^{(2)}, \ldots, x^{(m)}]$ is then independently processed by our focused RNNs layer. The number of focused downstream RNNs n is an important hyper-parameter that depends on the language. For each word embedding $x^{(t)}$ the filtering RNN (RNN^{filter}) first produces a vector of attention weights $f^{(t)}$. The filtering/gating mechanism multiplies each weight $f_i^{(t)}$ with the same word embedding $x^{(t)}$ to produce a weighted word embedding $a_i^{(t)}$, where $i \in [1, n]$. The weighted sequence of word embeddings produced in this way makes it possible for downstream RNNs (RNN_i) to focus on different aspects and project each one into a fixed-size vector representation in an argument embedding subspace b_i. Afterwards, these vectors are concatenated into a longer vector y that represents the final sentence/argument embedding for each text span. Final sentence/argument embeddings are computed as

$$\begin{aligned} y &= FR(x) \\ y' &= FR(x') \\ y'' &= FR(x'') \\ y''' &= FR(x''') \end{aligned} \tag{4.2}$$

where $FR(\cdot)$ is a function representing the focused RNNs layer, x is the sequence of word embeddings for `arg1`, x' for `arg2`, x'' for `conn`, x''' for `punc`, y is the argument embedding for `arg1`, y' for `arg2`, y'' for `conn`, and y''' for `punc`.

We apply the same focused RNNs layer to all four text spans, i.e. share their trainable weights. This encourages that information on different aspects of text spans is aggregated into argument embeddings y in the same vector space, instead of overfitting on specifics of each text span.

4.2.3 Feed-forward classification layer

All argument embeddings are concatenated into a longer vector c and passed into a two-layer *feed-forward neural network* (FFNN) to predict the probabilities p for sense classification.

First, all argument embeddings are processed by a feed-forward layer with the SReLU activation function, afterwards another feed-forward layer with the Softmax activation function is put on top to compute the classification probability distribution p as

$$\begin{aligned} c &= y \| y' \| y'' \| y''' \\ d &= SReLU(\mathrm{W}_D c + b_D) \\ p &= Softmax(\mathrm{W}_P d + b_P) \end{aligned} \tag{4.3}$$

where operator $\|$ represents concatenation of vectors, $SReLU(\cdot)$ and $Softmax(\cdot)$ represent the corresponding activation functions (details are described below), W_D, W_P, b_D, and b_P are trainable parameters of feed-forward layers, y is the argument embedding for `arg1`, y' for `arg2`, y'' for `conn`, and y''' for `punc`.

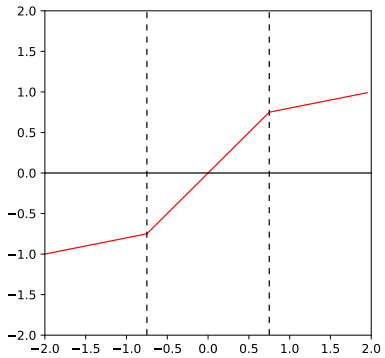
The S-shaped rectified linear activation unit (SReLU) [52] consists of a piecewise linear function with three parts. It is defined as

$$SReLU(z_i) = \begin{cases} t_i^L + a_i^L(z_i - t_i^L) & z_i \le t_i^L \\ z_i & t_i^L < z_i < t_i^R \\ t_i^R + a_i^R(z_i - t_i^R) & t_i^R \le z_i \end{cases} \tag{4.4}$$

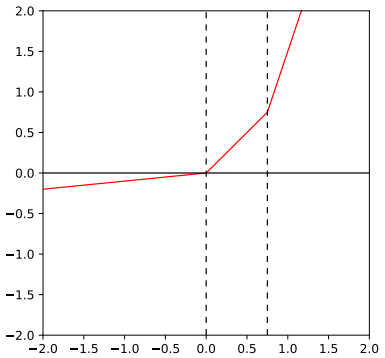
where t_i^L and a_i^L are the trainable left threshold and slope, t_i^R and a_i^R are the trainable right threshold and slope, and the subscript i indicates that we allow SReLU to vary in different dimensions of its input vectors. Due to its construction it is capable of learning both convex and non-convex functions (see Fig. 4.2), but it is faster to compute than traditional trigonometric functions. ReLU, LReLU, PReLU, and similar activation functions can be seen as special cases of SReLU. This makes the SReLU activation function perform somewhat better for sense classification than the convex-only activation function or without it.

The Softmax activation function or transformation computes a probability vector over its inputs, like a logistic regression. It is defined as

$$Softmax(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \tag{4.5}$$



(a) SReLU imitating a hyperbolic tangent function.
$t^L = -0.75, a^L = 0.2, t^R = 0.75, a^R = 0.2$



(b) SReLU imitating a leaky rectified linear unit (LReLU).
$t^L = 0.0, a^L = 0.1, t^R = 0.75, a^R = 3.0$

Figure 4.2
SReLU activation function can approximate other activation functions by learning the correct parameters. SReLU can learn both non-linear (b) convex and (a) non-convex functions.

Finally, FR system returns the sense label with the highest probability p as the result of sense classification for the given discourse relation.

4.3 Implementation of the FR system

The FR system for sense classification is publicly available on http://github.com/gwo/conll16st-v35-focused-rnns/ under the AGPL-3.0+ license. It is implemented in Python 2.7 using the Keras 1.2.2 library [53]. The Keras library provides a high-level API for developing neural networks, it is based on Theano and TensorFlow, and is capable of running on either CPU or GPU. All models and their training procedures are implemented in Keras.

Training can be performed in an end-to-end manner with backpropagation and any gradient-based optimization algorithm, because all operations involved in are fully differentiable. We chose to use the Adam optimizer [54] because it is well suited for problems that incorporate many parameters. To parallelize and speed up the learning process, training is done in mini-batches of 64 training samples. For efficiency reasons, we unroll all LSTM layers to the maximal size of each text span and use a fixed mini-batch dimension. Because text spans have a variable length, we use the masking technique to prevent unneeded computation. In addition to tracking the loss function on the train and valid

datasets, we also periodically evaluate the performance of our model on the valid dataset using the official evaluation methodology of CoNLL 2016 Shared Task. We stop the training procedure when there is no more improvement on the valid dataset in the last 10 epochs, i.e. ten passes through all the training samples.

Sense classification is a supervised multi-class classification task where a probabilistic interpretation of results is desired. A suitable training objective is therefore the categorical cross-entropy loss function, also known as multi-class log loss. The goal of training is to minimize the difference between the computed approximating distribution p and the one-hot vector encoding of the actual sense label.

We analyse the training and classification time complexity of the FR system in Section 5.4.

4.3.1 Data augmentation

During training we perform a simple data augmentation technique to make the model more robust to noise and improve the learning of task-specific word or character embeddings from scratch. We transform each original discourse relation in the training dataset into 2 positive and 2 negative samples.

For positive samples the sense label remains the same, because we introduce only so little noise that it should not affect the overall meaning. This improves the robustness of the classifier with respect to noise in data. For positive samples there is a 30% probability that 10% of symbols in `arg1` and `arg2` get mutated by each of the following three functions. Let us illustrate their effects in brackets on an example at word level (`it may be important`) and at character level (`then`):

- duplicate a randomly chosen symbol (`it may may be important`) (`theen`)
- insert an out-of-vocabulary symbol at random (`it may [OOV] be important`) (`the[OOV]n`)
- forget a randomly chosen symbol (`it be important`) (`thn`)

For negative samples we use a special no-sense label, because at least a part of them is always replaced with random symbols from the vocabulary, thus the text itself does not make any sense and there is no discourse relation anymore. This also improves the robustness of intermediate representations and counteracts the need to normalize word

or character embeddings on the whole vocabulary. For negative samples conn and punc are always replaced with random symbols of same length. Afterwards there is a 70% probability that arg1 and arg2 get mutated by each of the following three functions (and in the unlikely event that nothing changed, everything is replaced with random symbols of maximal length):

- replace arg1 with random symbols of same length (`crash $ Charles 20.9`) (`nL3t`)
- replace arg2 with random symbols of same length
- swap arg1 and arg2

4.3.2 *Hyper-parameters*

We use the same model hyper-parameters on all settings to make it usable almost out-of-the-box on any language, set of sense labels, or level of input representation. In the fine-tuning process we started with a reasonably-working setting and attempted to tune each hyper-parameter with a local grid search individually. All hyper-parameters and other design choices for each setting were fit by using the train and valid datasets for both languages. It is interesting to note that attempts at fine-tuning the model hyper-parameters for different settings did not substantially improve its performance in F_1-scores. Due to the differences in average sentence and token lengths, and number of sense labels, a few basic hyper-parameters had to be adjusted as described in Tab. 4.1. Due to practical reasons (memory consumption, large training times), we truncate arguments that are too long (m, m', m'', m''') in such a way that less then 1% of samples are affected (see the distribution of argument lengths in Fig. 2.1). The optimal number of focused downstream RNNs depends on the language and not on whether it is applied at word or character level. All other parameters should use the values described in this subsection.

As can be seen from Tab. 4.1 it is sufficient to have the dimensionality of the word embedding layer only $|x^{(t)}| = 20$, of the filtering RNN layer $|f^{(t)}| = 8$ for English and $|f^{(t)}| = 12$ for Chinese (to match the number of focused downstream RNNs n), of individual downstream RNNs $|b_i| = 20$, and of the FFNN hidden layer 80. Our goal is to predict only sense labels, there are 11 for Chinese and 22 for English, including partially annotated sense labels and a special no-sense label, which is used for negative samples.

Table 4.1
Adjusted basic model hyper-parameters for English and Chinese languages.

		Chinese			English		
		FR-zh	FR-zhch	simple-zh	FR-en	FR-ench	simple-en
Parameter							
- max arg1 length	m	500	900	500	100	400	100
- max arg2 length	m'	500	900	500	100	400	100
- max conn length	m''	10	20	10	10	20	10
- max punc length	m'''	2	2	2	0	0	0
- num. focused RNNs	n	12	12	–	8	8	–
Dimensionality							
- word emb. dim.	$\|x^{(t)}\|$	20	20	20	20	20	20
- filtering RNN dim.	$\|f^{(t)}\|$	12	12	–	8	8	–
- focused RNNs dim.	$\|b_i\|$	20	20	–	20	20	–
- simple LSTMs dim.		–	–	240	–	–	160
- FFNN input dim.	$\|c\|$	960	960	960	480	480	480
- sense labels	$\|p\|$	11	11	11	22	22	22
Trainable weights							
- word emb. layer		295,700	57,180	295,700	878,360	1,740	878,360
- other layers		120,700	120,700	1,013,772	68,759	68,759	358,583

FR-zh and FR-en represent FR system at the word level, FR-zhch and FR-ench represent FR system at the character level, simple-zh and simple-en represents a strong baseline model with simple LSTMs.

Initial values of trainable weights are set according to best practices, as they do not affect the training outcome substantially. The word embedding layer is therefore initialized with a uniform random distribution, input transformation of all LSTMs with Glorot uniform random distribution [55], and transformations of their recurrent state with an orthogonal matrix, and all FFNN layers and the slopes of the SReLU activation function again with Glorot uniform random distribution.

Due to many trainable parameters and the lack of training samples, we improve the generalizability of our model with dropout layers [56] and sharing of trainable parameters. Dropout is a well-known regularization technique that reduces overfitting in neural networks by preventing complex co-adaptations in the training dataset. We introduce dropout layers with 0.3 fraction of entries that will be randomly set to 0 at each update during training time. We add them after each major layer of our model: after the word embedding layer $x^{(t)}$, after the concatenated argument embeddings of focused RNNs layer c, and after the FFNN hidden layer before classification d. We also performed

some experiments with dropout and zoneout regularization on recurrent connections of RNNs, but there were no substantial improvements. Furthermore, we tried to introduce curriculum learning by gradually increasing the length of arguments during training but again, with no substantial improvements. To improve on generalizability, our model also performs sharing of trainable parameters for word embeddings layer, filtering RNN, and focused downstream RNNs, as described in previous subsections. Experiments have shown that disabling the sharing of trainable parameters degrades the performance of our model for sense classification.

5

Evaluation

In this chapter we evaluate our FR system on Chinese and English. We first analyse its overall performance in terms of F_1-score and Brier-score, then in more detail with per-sense results and confusion matrices for `Explicit` and non-`Explicit` relations, and perform a case study of errors on English. We also analyse its training and classification time complexity. To qualitatively assess the contribution of some design choices we also perform an ablation study.

5.1 Methodology

The CoNLL 2016 Shared Task [16], organized within the CoNLL conference, provides an official task formulation, datasets, and evaluation methodology for sense classification of shallow discourse relations. We presented the task definition in Section 1.2 and described the Chinese and English datasets in Section 2.1.1 and 2.1.2. Related work for sense classification follows the official methodology and evaluates its performance in terms of F_1-score on these datasets. We also primarily follow the official methodology to be able to compare the FR system with winning systems of the CoNLL 2016 Shared Task and strong baselines. We then go a step further and also explore the model and its results with other evaluation techniques.

Official datasets consist of four datasets for each language. The train dataset is the largest and contains training samples for model fitting. The valid or development dataset contains samples used for tuning model hyper-parameters and guiding design decisions. The test dataset is used for evaluation on the same corpus as train dataset. The blind test dataset is used for evaluation on a slightly out-of-domain corpus. The official ranking is based on the blind dataset, because it evaluates the robustness of the model and its performance in practical situations. The last dataset is called blind dataset, because it was hidden from the participants of the competition. The participants had to deploy their complete systems on a remote evaluation system that was disconnected from the internet when computing the official ranking. The official datasets also provide additional layers of automatic linguistic annotation processed with state-of-the-art NLP tools (POS tags, syntactic parse trees, and dependency parses), but in our approach we ignore this information.

In general, a statistical approach for comparing the performance of different methods is to evaluate them on a large set of datasets, compute the critical difference between scores, and test the statistical significance of results. Unfortunately, for sense classifica-

tion there are only two sufficiently large and complete datasets available and even those are for two different languages, one for Chinese and one for English. Most other methods are also language-dependent, so we only have one performance measurement for most methods. It is therefore impossible to talk about the significance of our results, but we can still explore the method in the following ways.

F_1-*score* is the primary metric for evaluating the performance of sense classifiers (higher score is better). The official ranking for CoNLL 2016 Shared Task is based upon the overall results on all relations (`All`) on the blind dataset. F_1-score is based on the number of discourse relations where the predicted sense label and actual sense label match exactly. In cases where the actual discourse relation is annotated with two or more sense labels the predicted sense label must match one of these sense labels to be considered correct. In cases where the actual sense label is only partially annotated, the predicted sense must match the partially annotated sense (although the blind datasets do not contain partial annotation). F_1-score calculates the harmonic mean of precision-score P and recall-score R as

$$\begin{aligned} F_1 &= 2 \cdot \frac{P \cdot R}{P + R} \\ P &= \frac{\sum \textit{True positives}}{\sum \textit{True positives} + \sum \textit{False positives}} \\ R &= \frac{\sum \textit{True positives}}{\sum \textit{True positives} + \sum \textit{False negatives}} \end{aligned} \tag{5.1}$$

First, we analyse the *overall results* in F_1-score on the valid, test, and blind datasets for all (`All`), `Explicit` (`Exp`), and non-`Explicit` (`Non-E`) relations. These are computed as multi-class micro-averaged F_1-score of all sense labels. Because we have a prediction for each discourse relation and iterate through all sense labels, it can be shown that the F_1-score is the same as classification accuracy. Next, we analyse the *per-sense results* in F_1-score on the valid, test, and blind datasets for `Explicit` and non-`Explicit` relations. These enable us to compare the performance for each sense label with other methods and also how it changes for `Explicit` and non-`Explicit` relations.

With the *Brier-score* we perform an additional evaluation to determine the error of probabilistic predictions p produced by our method (lower score is better). Namely, the F_1-score evaluates only the discrete outcomes (only one sense label is predicted and it is either correct or not). On the other hand, our method internally assigns probabilities to all sense labels and Brier-score is an appropriate measure to evaluate their correctness.

Because related work did not evaluate Brier-scores, we can only compare our method with strong baselines. Brier-score calculates the mean squared error/difference between the predicted probabilities p and actual outcomes o over all N sense labels as

$$Brier = \frac{1}{N} \cdot \sum_{j=1}^{N} (p_j - o_j)^2 \tag{5.2}$$

We use a *confusion matrix* to explore the results of our model in more detail. It is a matrix that counts the number of occurrences, where each row represents a predicted sense label, and each column the actual sense label. The diagonal of a confusion matrix counts the number of occurrences the predicted and actual sense label matched, while off-diagonal elements count the errors. A confusion matrix is typically used to identify the common types of mistakes a method makes. Because of the convenient location we also add to the matrix the total number of occurrences, precision-score P, and recall-score R. Because the distribution of sense labels is highly imbalanced, any normalization attempts would skew the results and therefore we do not attempt to visualize it as a heatmap.

We continue with a *case study of particular errors* on English. We provide a few examples of discourse relations with a particular sense label and display some probabilities produced by our method. This way the reader can get an even better understanding of the errors and evaluate by himself how difficult they really are.

In Section 5.4 we analyse the training and classification *time complexity* of the FR system. First we determine the theoretical time complexity, then we visualize the empirical time complexity in different settings.

The most important and unique characteristic of our FR system is that it can be used almost out-of-the-box on any language, set of sense labels, or level of input representation. To confirm its language-independence we apply almost the same model hyper-parameters on two substantially different languages. On Chinese, as an example of a less supported language, and on English, as the language with most research and advanced language technologies. To qualitatively assess the contribution of some design choices we also perform an *ablation study*.

5.2 Results on Chinese

We compare our FR system for sense classification with the following winning systems of the CoNLL 2016 Shared Task [16] and strong baselines for Chinese:

- *FR-zh*: Our FR system for Chinese at the word level (previously FR-wa).
- *FR-zhch*: Our FR system for Chinese at the character level (previously FR-ca).
- *Weiss & Bajec [22]*: The previous overall best sense classifier for Chinese. This is our older two-model system that received the first award by a large margin on the CoNLL 2016 Shared Task. It uses two separate models with focused RNNs at word level, many fine-tuned parameters, and trains with random noise samples, but uses no external resources.
- *Wang & Lan [21]*: The best discourse parser for Chinese (with sense classification results). For `Explicit` relations it uses a logistic regression classifier on many hand-engineered features based on connectives and their context, and for non-`Explicit` relations production rules and features with word and verb pairs at specific locations. It uses POS tags, parse trees, and word categories.
- *Schenk et al. [28]*: The second best `Explicit` sense classifier for Chinese. For `Explicit` relations it uses a SVM classifier only on the connectives, and for non-`Explicit` relations a series of summations and multiplications of word and parse tree embeddings. It uses pre-trained word embeddings and parse trees.
- *random-zh*: A minimal baseline model for Chinese that returns sense labels uniformly at random.
- *majority-zh*: A minimal baseline model for Chinese that returns only the most common sense label (i.e. Conjunction).
- *simple-zh*: A strong baseline model for Chinese similar to our FR system, but with a single 240-dimensional LSTM layer for each text span instead of the focused RNNs layer. Due to it having more than three times as many trainable weights than our model (see Tab. 4.1), it should be far more powerful. It trains with our data augmentation, but uses no external resources.

Table 5.1

Overall results in F_1-score on Chinese datasets. Higher scores on the blind dataset indicate better performance and were used for the official ranking.

Models	valid			test			blind		
	All	Exp	Non-E	All	Exp	Non-E	All	Exp	Non-E
FR system									
- word level (FR-zh)	0.7520	0.9351	0.7059	0.7363	0.9375	0.6825	0.7396	0.7597	**0.7322**
- char level (FR-zhch)	0.7415	0.9351	0.6928	0.7253	0.9271	0.6713	**0.7477**	**0.8463**	0.7114
Prior work									
- Weiss & Bajec [22] (older)	0.7206	0.9351	0.6667	0.7011	0.9271	0.6407	0.7292	0.7898	0.7068
- Wang & Lan [21]	0.7807	0.9610	0.7353	0.7701	0.9424	0.7242	0.6473	0.7669	0.6052
- Schenk et al. [28]	0.7572	0.9610	0.7059	0.7701	0.9634	0.7187	0.6373	0.8039	0.5759
Baseline models									
- random-zh	0.0992	0.0909	0.1013	0.1121	0.0729	0.1226	0.0995	0.0883	0.1036
- majority-zh	0.5770	0.4156	0.6176	0.6110	0.5208	0.6351	0.5788	0.2880	0.6860
- simple-zh	0.7363	0.9221	0.6895	0.7231	0.8854	0.6797	0.6921	0.7968	0.6534
Ablation study									
- word level & word2vec	0.7493	0.9481	0.6993	0.7297	0.9479	0.6713	0.7373	0.7827	0.7205
- word level & ¬ augm.	0.7285	0.9481	0.6732	0.7429	0.9271	0.6936	0.7120	0.7951	0.6814
- word level & ¬ shared	0.7389	0.9610	0.6830	0.7297	0.9583	0.6685	0.7301	0.7915	0.7075
- word level & ¬ dropout	0.7076	0.9221	0.6536	0.7319	0.9271	0.6797	0.6782	0.7792	0.6410
- word level & GRU	0.7076	0.9221	0.6536	0.7121	0.9167	0.6574	0.7211	0.7138	0.7238
- word level & dim.×2	0.7337	0.9610	0.6765	0.7473	0.9583	0.6908	0.7192	0.7809	0.6964
- word level & dim.÷2	0.7232	0.9481	0.6667	0.7121	0.9479	0.6490	0.7368	0.7739	0.7231
- char level & ¬ augm.	0.7180	0.8961	0.6732	0.7253	0.9271	0.6713	0.7454	0.7862	0.7303
- char level & ¬ shared	0.7363	0.9481	0.6830	0.6923	0.9167	0.6323	0.6787	0.8357	0.6208

We first evaluate the performance of all systems in terms of the overall results in F_1-score in Tab. 5.1 and in Brier-score in Tab. 5.2 on Chinese datasets. F_1-score is the primary metric for evaluating the performance of sense classifiers (higher score is better), so we can compare it with reported results of prior work and strong baselines. Brier-score is used to evaluate the error of probabilistic predictions (lower is better) that was not used by prior work, so we can only compare our method with strong baselines.

Table 5.2

Overall results in Brier-score on Chinese datasets. Lower scores on the blind dataset indicate better performance.

	valid			test			blind		
Models	All	Exp	Non-E	All	Exp	Non-E	All	Exp	Non-E
FR system									
- word level (FR-zh)	0.3787	0.1529	0.4355	0.3514	0.0941	0.4203	**0.4123**	0.3613	**0.4311**
- char level (FR-zhch)	0.3757	0.1043	0.4440	0.3524	0.1138	0.4162	0.4174	**0.2259**	0.4881
Baseline models									
- random-zh	1.8016	1.8182	1.7974	1.7758	1.8542	1.7549	1.8010	1.8233	1.7928
- majority-zh	0.8460	1.1688	0.7647	0.7780	0.9583	0.7298	0.8425	1.4240	0.6280
- simple-zh	0.4231	0.1635	0.4884	0.3857	0.1339	0.4530	0.4652	0.3228	0.5177
Ablation study									
- word level & wordzvec	0.4041	0.0966	0.4815	0.3843	0.0802	0.4657	0.4102	0.3261	0.4412
- word level & ¬ augm.	0.4121	0.1000	0.4907	0.3599	0.0950	0.4308	0.4324	0.2861	0.4864
- word level & ¬ shared	0.4076	0.0784	0.4905	0.3909	0.0848	0.4728	0.4093	0.2944	0.4517
- word level & ¬ dropout	0.4622	0.1510	0.5405	0.3742	0.1058	0.4459	0.4874	0.3122	0.5520
- word level & GRU	0.4437	0.2221	0.4995	0.3903	0.1587	0.4523	0.4327	0.4351	0.4318
- word level & dim.×2	0.4064	0.0812	0.4882	0.3552	0.0907	0.4259	0.4145	0.2995	0.4569
- word level & dim.÷2	0.4238	0.1330	0.4970	0.3939	0.0843	0.4767	0.4038	0.3313	0.4305
- char level & ¬ augm.	0.4046	0.1711	0.4634	0.3655	0.1528	0.4223	0.4161	0.2867	0.4638
- char level & ¬ shared	0.3990	0.0926	0.4761	0.4126	0.1082	0.4940	0.4565	0.2221	0.5429

The official ranking for CoNLL 2016 Shared Task is based upon the *overall results in* F_1*-score* for all relations (All) on the blind dataset, which is presented in Tab. 5.1. Both our systems (FR-zh, FR-zhch) outperform all other systems on the blind dataset. The FR-zhch system even outperforms our older two-model system [22] by 2.5% for all relations. In comparison to other systems not using the focused RNNs layer, it improves by even more than 8%. One would expect that more complex and fine-tuned systems for a language outperform systems with only a single end-to-end trainable model. For the FR-

zhch system the performance increase comes from the much higher results for `Explicit` relations (`Exp`). It even outperforms the previous best system by Schenk et al. [28], which uses a simpler model for `Explicit` relations. Having a simpler model might be beneficial in Chinese, because there are far fewer `Explicit` training samples available. For non-`Explicit` relations, it is interesting to note that the performance of most approaches on the blind dataset is far below the majority-zh baseline. This clearly suggests that Schenk et al. [28] and Wang & Lan [21] overfit the training domain and style of the CDTB corpus. On the other hand, both our systems (FR-zh, FR-zhch) capture the target concepts better. The FR-zh system even outperforms our older two-model system [22] by 3.6% for non-`Explicit`relations.

Analog to the official ranking, it makes sense to analyse the *overall results in Brier-score* for all relations (`All`) also on the blind dataset, which is presented in Tab. 5.2. Both our systems (FR-zh, FR-zhch) again outperform all strong baselines on the blind dataset. However, the ranking does not stay the same, because we are now evaluating the error of probabilistic predictions. Now the FR-zh system performs slightly better than FR-zhch for all relations, mainly because of its performance for non-`Explicit` relations that occur far more often.

We also perform an *ablation study* to qualitatively assess the contribution of some design choices (results are in bottom of Tab. 5.1 and Tab. 5.2). First one, combines the FR-zh system with pre-trained word embeddings (& word2vec). We initialize the word embeddings layer with 300-dimensional embeddings produced by the Skip-gram model from Word2vec [41] on the Gigaword simplified Chinese dataset. Second, uses the FR-zh system without our simple data augmentation technique (&¬ augm.) and performs training only on positive samples. Third, uses the FR-zh system without sharing of trainable parameters (& ¬ shared). Forth, uses the FR-zh system without dropout regularization technique (& ¬ dropout). Fifth, replaces in the FR-zh system all LSTMs with Gated-recurrent units (GRUs) [57] in the focused RNNs layer (& GRU). Sixth, uses the FR-zh system with all dimensions multiplied by 2 (& dim.×2). Seventh, uses the FR-zh system with all dimensions divided by 2 (& dim.÷2). Eight, uses the FR-zhch system without our simple data augmentation technique (& ¬ augm.). Ninth, uses the FR-zhch system without sharing of trainable parameters (& ¬ shared). The results indicate that the design choices for both of our systems (FR-zh, FR-zhch) are near a local optimum. Contrary to expectations, introducing pre-trained word embeddings (& word2vec) does not seem to substantially improve the performance. This suggests that

the same semantic and syntactic information relevant for sense classification on Chinese can also be learned from scratch. Disabling either sharing of trainable parameters (& ¬ shared) or dropout regularization technique (& ¬ dropout) notable degrades the performance on non-`Explicit` relations. Disabling data augmentation technique (&¬ augm.) only degrades the performance when used at word level.

Previous studies [30] suggest that there is a substantial difference between `Explicit` and non-`Explicit` relations, thus we continue with a detailed analysis of both situations on Chinese datasets. In Tab. 2.2 we see that Chinese datasets are small, have a highly imbalanced distribution of sense labels, and many sense labels only have a few training samples. These probably act more as noise than contribute to the learning. Merging them into one target class for sense classification would reformulate the task, but on the other hand probably improve the overall performance. We reject the idea of manually manipulating with target classes, because it is in conflict with our ambition of not using any hand-engineering.

5.2.1 Analysis of `Explicit` relations

We continue with an analysis of per-sense results in F_1-score in Tab. 5.3 and confusion matrix in Tab. 5.4 for `Explicit` relations on Chinese datasets. Per-sense results enable us to compare the performance for each sense label with other methods. Confusion matrix allows us to explore the results and identify the common types of mistakes made.

We analyse the *per-sense results in* F_1*-score for `Explicit` relations* primarily on the blind dataset, which is presented in Tab. 5.3. As expected, the results on the valid and test datasets are better because they originate from the same CDTB corpus as the train dataset. On the other hand, on the slightly out-of-domain blind dataset we see a degradation of more than 10% for all methods for two most common sense labels (Conjunction and Expansion). This suggests that they are realized differently in the blind dataset, and manual feature engineering to disambiguate their meaning could substantially improve the results. Nevertheless, we see that the FR-zhch system achieves the best results with a large margin for more common sense labels in the train dataset, especially Conjunction, Contrast and Purpose. The FR-zh system still achieves competitive overall performance. Our strong baselines model simple-zh performs surprisingly well for Conditional. The strength of incorporating linguistic knowledge and hand-engineered features into a system, as in Wang & Lan [21], is reflected in better performance for sense labels with only a few samples, such as Alternative and Progression.

Table 5.3

Per-sense results in F_1-score for `Explicit` relations on Chinese datasets.

	valid		test		blind				
Sense label	FR-zh	FR-zhch	FR-zh	FR-zhch	FR-zh	FR-zhch	[22]	[21]	[28]	simple-zh
Alternative	–	–	–	–	0.	0.	0.	**0.1000**	0.0952	0.
Causation	1.0000	1.0000	1.0000	1.0000	0.8850	**0.9524**	0.9434	0.9216	**0.9524**	0.9434
Conditional	1.0000	0.9091	0.8000	0.8000	0.7294	0.8077	0.8454	0.8791	0.8866	**0.9231**
Conjunction	0.9275	0.9412	0.9615	0.9495	0.7930	**0.8515**	0.7726	0.7324	0.7711	0.7673
Contrast	0.8750	0.8750	1.0000	0.8571	0.7482	**0.8452**	0.7564	0.7245	0.7571	0.7639
EntRel	–	–	–	–	–	–	–	–	–	–
Expansion	1.0000	1.0000	0.8000	0.9333	0.5714	0.7500	0.7727	0.7556	**0.7907**	0.7273
Progression	0.	0.	0.	0.	0.	**0.3333**	0.	**0.3333**	0.2857	**0.3333**
Purpose	1.0000	1.0000	1.0000	1.0000	0.6667	**0.9474**	0.8182	0.9000	0.9000	0.8571
Temporal	0.9412	0.9412	0.9091	0.9143	0.8143	0.9259	0.8659	0.8591	**0.9299**	0.8774
Overall	0.9351	0.9351	0.9375	0.9271	0.7597	**0.8463**	0.7898	0.7669	0.8039	0.7968

We also explore the *confusion matrix for `Explicit` relations* from the FR-zh system on the blind dataset, which is presented in Tab. 5.4. Because the confusion matrices for both our systems (FR-zh, FR-zhch) look similar, we present only one of them. Two sense labels (EntRel and Progression) occur less than 5 times, so we merged their corresponding rows and columns to make the confusion matrix clearer. The highest

Table 5.4

Confusion matrix for `Explicit` relations from the FR-zh system on Chinese blind dataset. Bold indicates counts that occur at least one-fourth of the time.

Predicted \ Truth	Alternative	Causation	Conditional	Conjunction	Contrast	Expansion	Purpose	Temporal	Other (2 sense labels)	Total	Precision
Alternative	0	–	–	–	–	–	–	–	–	0	1.0000
Causation	–	**50**	5	–	3	1	–	–	–	59	0.8475
Conditional	1	–	**31**	1	3	–	–	–	–	36	0.8611
Conjunction	**6**	3	4	**159**	37	4	–	**22**	3	238	0.6681
Contrast	**5**	–	–	1	**104**	–	–	–	–	110	0.9455
Expansion	3	–	2	–	19	**20**	–	1	–	45	0.4444
Purpose	–	1	6	–	2	–	**9**	–	–	18	0.5000
Temporal	–	–	1	2	–	–	–	**57**	–	60	0.9500
Other (2 sense labels)	–	–	–	–	–	–	–	–	0	0	–
Total	15	54	49	163	168	25	9	80	3		
Recall	0.0000	0.9259	0.6327	0.9755	0.6190	0.8000	1.0000	0.7125	–		

recall-score is achieved by Conjunction, because our method has a bias to classify most samples with this sense label. This is probably an artifact of Conjunction being the most common sense label in the train dataset (see Tab. 2.2). Unfortunately, it also means that its precision-score is lower. It is interesting to note, that Contrast and Temporal are confused with Conjunction more than one-fifth of the time. The highest precision-score is achieved for Contrast, probably because discourse connectives for Contrast have only one meaning.

5.2.2 *Analysis of non-Explicit relations*

We continue with an analysis of per-sense results in F_1-score in Tab. 5.5 and confusion matrix in Tab. 5.6 for non-Explicit relations on Chinese datasets. Non-Explicit relation types are Implicit, AltLex, and EntRel (described in Section 2.1).

We analyse the *per-sense results in F_1-score for non-Explicit relations* primarily on the blind dataset, which is presented in Tab. 5.5. The distribution of sense labels for Chinese non-Explicit relations is highly imbalanced. The most common sense, Conjunction, occurs approximately 5-times more frequently than the second and the third sense label. Therefore, the overall results are highly correlated with the performance on Conjunction. All systems based on focused RNNs layer perform much better on the sense Conjunction and therefore substantially outperform other systems. In addition to that, the FR-zh system performs slightly better on Contrast and Expansion, which makes it the overall best performing system. Although the train dataset contains many samples of EntRel relations, our method seems incapable of automatically learning the related concept of coreferent entity mentions. On the other hand, the hand-engineered system by Wang & Lan [21] outperforms on two very low-frequent sense labels, Alternative and Progression, but fails on most more-frequent ones, especially Causation and Conjunction. Overall, the performance of our systems improves on state-of-the-art, despite using only a single end-to-end trainable model, no hand-engineered features or external resources.

We also explore the *confusion matrix for non-Explicit relations* from the FR-zh system on the blind dataset, which is presented in Tab. 5.6. Because the confusion matrices for both our systems (FR-zh, FR-zhch) look similar, we present only one of them. Two sense labels (Alternative and Progression) occur less than 5 times, so we merged their corresponding rows and columns to make the confusion matrix clearer. The highest recall-score is achieved by Conjunction, because our method has a bias to classify most

Table 5.5

Per-sense results in F_1-score for non-`Explicit` relations on Chinese datasets.

	valid		test		blind				
Sense label	FR-zh	FR-zhch	FR-zh	FR-zhch	FR-zh	FR-zhch	[22]	[21]	[28]	simple-zh
Alternative	–	–	–	–	0.	0.	0.	**0.5000**	**0.5000**	0.
Causation	0.2667	0.1538	0.2500	0.3529	0.2333	**0.2712**	0.1754	0.0755	0.0392	0.1481
Conditional	0.	1.0000	0.	0.	0.	0.	0.	0.	0.	0.
Conjunction	0.8198	0.8035	0.8068	0.8022	**0.8388**	0.8278	0.8213	0.7442	0.7294	0.7843
Contrast	0.	0.	0.	0.	0.1200	0.0800	0.0784	0.0408	**0.1481**	0.0816
EntRel	0.3714	0.0392	0.3301	0.	0.1449	0.	0.	**0.2090**	0.1982	0.1926
Expansion	0.4800	0.5926	0.4000	0.4516	**0.5455**	0.3825	0.5250	0.5024	0.4387	0.4171
Progression	–	–	–	–	0.	0.	0.	**0.3333**	0.2857	0.
Purpose	0.6667	0.6667	0.	0.	0.1333	0.0690	**0.3333**	0.1250	0.1250	0.2857
Temporal	–	–	0.6667	1.0000	0.3333	0.4211	0.3333	0.3000	0.3636	**0.4800**
Overall	0.7059	0.6825	0.6825	0.6713	**0.7322**	0.7114	0.7068	0.6052	0.5759	0.6534

samples with this sense label. This is probably an artifact of Conjunction being by far the most common sense label in the train dataset (see Tab. 2.2). Unfortunately, it also means that its precision-score is lower. It is interesting to note, that all sense labels are almost always confused with Conjunction when discourse connectives are not explicitly expressed.

Table 5.6

Confusion matrix for non-`Explicit` relations from the FR-zh system on Chinese blind dataset. Bold indicates counts that occur at least one-fourth of the time.

Predicted \ Truth	Causation	Conditional	Conjunction	Contrast	EntRel	Expansion	Purpose	Temporal	Other (2 sense labels)	Total	Precision
Causation	7	–	1	–	–	1	3	–	–	12	0.5833
Conditional	–	0	–	–	–	–	–	–	1	1	0.0000
Conjunction	**39**	**8**	**991**	**39**	**70**	**124**	**23**	**11**	5	1310	0.7565
Contrast	–	–	–	3	2	–	–	–	–	5	0.6000
EntRel	1	–	26	1	10	12	–	1	–	51	0.1961
Expansion	1	–	35	2	5	**108**	–	–	–	151	0.7152
Purpose	–	–	–	–	–	–	2	–	–	2	1.0000
Temporal	–	–	–	–	–	–	–	3	–	3	1.0000
Other (2 sense labels)	–	–	–	–	–	–	–	–	0	0	–
Total	48	8	1053	45	87	245	28	15	6		
Recall	0.1458	0.0000	0.9411	0.0667	0.1149	0.4408	0.0714	0.2000	–		

5.3 Results on English

We compare our FR system for sense classification with the following winning systems of the CoNLL 2016 Shared Task [16] and strong baselines for English:

- *FR-en*: Our FR system for English at the word level (previously FR-wa).
- *FR-ench*: Our FR system for English at the character level (previously FR-ca).
- *Mihaylov & Frank [19]*: The best overall sense classifier for English. For `Explicit` relations it uses a predefined list of connectives, and for non-`Explicit` relations two logistic regression classifiers on argument embeddings and several cross-argument similarity features. It uses pre-trained word embeddings and POS tags.
- *Oepen et al. [26]*: The best discourse parser for English. It uses a complex pipeline of fine-tuned models with hand-engineered features and production rules.
- *Rutherford & Xue [20]*: The best non-`Explicit` sense classifier for English. It uses a pooling function on pre-trained word embeddings of each argument followed by a three-layer FFNN. It is specialized only for non-`Explicit` relations.
- *random-en*: A minimal baseline model for English that returns sense labels uniformly at random.
- *majority-en*: A minimal baseline model for English that returns the most common sense label (i.e. Expansion.Conjunction).
- *simple-en*: A strong baseline model for English similar to our FR system, but with a single 160-dimensional LSTM layer for each text span instead of the focused RNNs layer. Due to it having more than three times as many trainable weights than our model (see Tab. 4.1), it should be far more powerful. It trains with our data augmentation, but uses no external resources.

We first evaluate the performance of all systems in terms of the overall results in F_1-score in Tab. 5.7 and in Brier-score in Tab. 5.8 on English datasets. F_1-score is the primary metric for evaluating the performance of sense classifiers (higher score is better), so we can compare it with reported results of prior work and strong baselines. Brier-score is

Table 5.7

Overall results in F_1-score on English datasets. Higher scores on the blind dataset indicate better performance and were used for the official ranking.

Models	valid			test			blind		
	All	Exp	Non-E	All	Exp	Non-E	All	Exp	Non-E
FR system									
- word level (FR-en)	0.6072	0.9059	0.3445	0.5819	0.8959	0.2962	0.5211	0.7680	0.3109
- char level (FR-ench)	0.6016	0.9014	0.3378	0.5550	0.8753	0.2636	0.5062	0.7230	0.3216
Prior work									
- Mihaylov & Frank [19]	0.6413	0.9120	0.4032	0.6331	0.8980	0.3919	**0.5460**	**0.7820**	0.3451
- Oepen et al. [26]	0.6570	0.9135	0.4312	0.6062	0.9013	0.3376	0.5356	0.7717	0.3384
- Rutherford & Xue [20]	–	–	0.4032	–	–	0.3613	–	–	**0.3767**
Baseline models									
- random-en	0.0717	0.0653	0.0774	0.0724	0.0857	0.0602	0.0736	0.0683	0.0781
- majority-en	0.2202	0.2807	0.1669	0.2088	0.2701	0.1530	0.2738	0.3903	0.1746
- simple-en	0.6229	0.9120	0.3685	0.5643	0.8796	0.2774	0.5012	0.7752	0.2680
Ablation study									
- word level & word2vec	0.6009	0.9029	0.3351	0.5886	0.8970	0.3080	0.5178	0.7662	0.3063
- word level & ¬ augm.	0.5788	0.8816	0.3124	0.5618	0.8764	0.2754	0.5062	0.7554	0.2940
- word level & ¬ shared	0.5746	0.8983	0.2897	0.5401	0.8839	0.2270	0.4673	0.7482	0.2282
- word level & ¬ dropout	0.5938	0.8938	0.3298	0.5747	0.8894	0.2883	0.4773	0.7518	0.2435
- word level & GRU	0.5859	0.8832	0.3244	0.5566	0.8677	0.2734	0.5079	0.7482	0.3032
- word level & dim.×2	0.6051	0.9105	0.3364	0.5659	0.8774	0.2823	0.4806	0.7338	0.2649
- word level & dim.÷2	0.5952	0.8907	0.3351	0.5711	0.8753	0.2942	0.5029	0.7428	0.2986
- char level & ¬ augm.	0.5845	0.8968	0.3097	0.5385	0.8753	0.2320	0.5070	0.7230	0.3231
- char level & ¬ shared	0.5881	0.8892	0.3231	0.5421	0.8590	0.2537	0.5021	0.7194	0.3170

used to evaluate the error of probabilistic predictions (lower is better) that was not used by prior work, so we can only compare our method with strong baselines.

The official ranking for CoNLL 2016 Shared Task is based upon the *overall results in* F_1*-score* for all relations (`All`) on the blind dataset, which is presented in Tab. 5.7. Both our systems (FR-en, FR-ench) do not fall a lot behind state-of-the-art performance on blind dataset by 4.6%, despite not using any linguistic knowledge or external resources. As expected, `Explicit` relations are classified better by Mihaylov & Frank [19], who use

Table 5.8

Overall results in Brier-score on English datasets. Lower scores on the blind dataset indicate better performance.

	valid			test			blind		
Models	All	Exp	Non-E	All	Exp	Non-E	All	Exp	Non-E
FR system									
- word level (FR-en)	0.5238	0.2487	0.7713	0.5251	0.2182	0.8038	0.5954	0.3233	0.8270
- char level (FR-ench)	0.5459	0.2407	0.8204	0.5376	0.2130	0.8324	0.6014	0.3553	**0.8110**
Baseline models									
- random-en	1.8566	1.8697	1.8451	1.8553	1.8286	1.8796	1.8528	1.8633	1.8438
- majority-en	1.5600	1.4394	1.6662	1.5824	1.4599	1.6940	1.4524	1.2194	1.6508
- simple-en	0.5117	0.2357	0.7599	0.5216	0.2168	0.7985	**0.5797**	**0.2958**	0.8214
Ablation study									
- word level & word2vec	0.5280	0.2276	0.7981	0.5281	0.2079	0.8189	0.5921	0.3126	0.8301
- word level & ¬ augm.	0.5607	0.2821	0.8114	0.5557	0.2576	0.8265	0.6013	0.3339	0.8291
- word level & ¬ shared	0.5388	0.2188	0.8266	0.5282	0.1893	0.8360	0.6140	0.3349	0.8516
- word level & ¬ dropout	0.5289	0.2297	0.7980	0.5303	0.2092	0.8220	0.6057	0.3221	0.8472
- word level & GRU	0.5672	0.2705	0.8340	0.5689	0.2443	0.8638	0.6385	0.3798	0.8587
- word level & dim.×2	0.5330	0.2395	0.7970	0.5408	0.2178	0.8342	0.6186	0.3530	0.8448
- word level & dim.÷2	0.5530	0.2863	0.7929	0.5438	0.2523	0.8086	0.6179	0.3707	0.8284
- char level & ¬ augm.	0.5655	0.2671	0.8339	0.5507	0.2287	0.8432	0.6073	0.3532	0.8236
- char level & ¬ shared	0.5506	0.2402	0.8297	0.5485	0.2304	0.8374	0.6094	0.3496	0.8307

a carefully defined list of discourse connectives and features connected with them. The FR-en system performs much better than FR-ench system for `Explicit` relations and consequently achieves higher scores. Because it works at the word level, it can easily learn more meaningful word embeddings for representing connectives. For non-`Explicit` relations, we see that that features used by Mihaylov & Frank [19], which are based on pre-trained word embeddings and cross-argument similarity, overfit the training domain and style of PDTB corpus. This method does not perform much better as ours on

the blind dataset. On the other hand, the specialized system by Rutherford & Xue [20] achieves better results. It uses a Bag-of-Words approach based on pre-trained word embeddings and the results suggest that the target concepts for the top six most common sense labels for English can be captured in word embeddings. However, substantially lower F_1-scores of all competing systems for English than for Chinese indicate that sense classification on English is much more difficult than on Chinese. Differences in the size of datasets, grammar, sense labels, and especially their distribution highly affect the performance.

Analog to the official ranking, it makes sense to analyse the *overall results in Brier-score* for all relations (`All`) also on the blind dataset, which is presented in Tab. 5.8. It is surprising that our strong baseline model simple-en performs in terms of Brier-score much better on `Explicit` relations and consequently for all relations. This means that it is capable of predicting the probabilities more correctly than both our systems (FR-en, FR-ench), but it often assigns the highest probability to the wrong sense label.

We also perform an *ablation study* to qualitatively assess the contribution of some design choices (results are in bottom of Tab. 5.7 and Tab. 5.8). First one, combines the FR-en system with pre-trained word embeddings (& word2vec). We initialize the word embeddings layer with 300-dimensional pre-trained word embeddings produced by the Skip-gram model from Word2vec [41] on the Google News English dataset. Second, uses the FR-en system without our simple data augmentation technique (&¬ augm.) and performs training only on positive samples. Third, uses the FR-en system without sharing of trainable parameters (& ¬ shared). Forth, uses the FR-en system without dropout regularization technique (& ¬ dropout). Fifth, replaces in the FR-en system all LSTMs with Gated-recurrent units (GRUs) [57] in the focused RNNs layer (& GRU). Sixth, uses the FR-en system with all dimensions multiplied by 2 (& dim.×2). Seventh, uses the FR-en system with all dimensions divided by 2 (& dim.÷2). Eight, uses the FR-ench system without our simple data augmentation technique (& ¬ augm.). Ninth, uses the FR-ench system without sharing of trainable parameters (& ¬ shared). The results indicate that the design choices for both of our systems (FR-en, FR-ench) are near a local optimum. Contrary to expectations, introducing pre-trained word embeddings (& word2vec) does not seem to substantially improve the performance. This suggests that the same semantic and syntactic information relevant for sense classification on English can also be learned from scratch. Disabling either sharing of trainable parameters (& ¬ shared) or dropout regularization technique (& ¬ dropout) degrades the performance,

especially for non-Explicit relations at word level. Disabling data augmentation technique (&¬ augm.) only degrades the performance when used at word level.

Previous studies [30] suggest that there is a substantial difference between Explicit and non-Explicit relations, thus we continue with a detailed analysis of both situations on English datasets. In Tab. 2.3 we see that the sense labels on English are unevenly distributed. Some sense labels, like Expansion.Exception, are only present in the train dataset and no other dataset, while 6 others contain merely a few training samples and are not even present in the blind dataset. These probably act more as noise than contribute to the learning. Merging them into one target class for sense classification would reformulate the task, but on the other hand probably improve the overall performance. We reject the idea of manually manipulating with target classes, because it is in conflict with our ambition of not using any hand-engineering.

5.3.1 Analysis of Explicit relations

We continue with an analysis of per-sense results in F_1-score in Tab. 5.9 and confusion matrix in Tab. 5.10 for Explicit relations on English datasets. Per-sense results enable us to compare the performance for each sense label with other methods. Confusion matrix allows us to explore the results and identify the common types of mistakes a method makes.

We analyse the *per-sense results in* F_1*-score for Explicit relations* primarily on the blind dataset, which is presented in Tab. 5.9. As expected, the results on the valid and test datasets are better, because they are from the same PDTB corpus as the train dataset. Although the train dataset contains many samples for sense labels Comparison.Contrast, Expansion.Instantiation and Temporal.Synchrony, we see a degradation of more than 15% in F_1-score for all systems on the slightly out-of-domain blind dataset. This suggests that they are realized differently in the blind dataset, and manual feature engineering to disambiguate their meaning could substantially improve the results. The FR-en system seems to learn the concept of sense label Contingency.Cause.Reason and Temporal.Synchrony slightly better than most other methods. It is competitive for Comparison.Contrast and Expansion.Conjunction, but falls behind in performance for other sense labels.

We also explore the *confusion matrix for Explicit relations* from the FR-en system on the blind dataset, which is presented in Tab. 5.10. Because the confusion matrices for both our systems (FR-en, FR-ench) look similar, we present only one of them. Six sense

Table 5.9

Per-sense results in F_1-score for `Explicit` relations on English datasets.

	valid		test		blind			
Sense label	FR-en	FR-ench	FR-en	FR-ench	FR-en	FR-ench	[19]	[26]	simple-en
Comparison.Concession	0.1250	0.2857	0.3810	0.2500	0.1000	0.0260	**0.2529**	0.1687	0.1463
Comparison.Contrast	0.9412	0.9532	0.9345	0.9250	0.3680	0.3662	**0.3934**	0.3680	0.3559
Contingency.Cause.Reason	0.8354	0.7500	0.9449	0.9194	0.8000	0.7200	0.7037	0.7719	**0.8438**
Contingency.Cause.Result	0.9143	0.8108	0.9589	0.8462	0.8462	0.5556	**0.9167**	**0.9167**	0.8462
Contingency.Condition	0.9333	0.9451	0.8793	0.8926	0.9455	0.9455	0.9455	0.9630	**0.9811**
EntRel	–	–	–	–	–	–	–	–	–
Expansion.Alt	0.8571	0.6667	0.8000	0.3810	0.6250	0.5263	**0.6667**	**0.6667**	**0.6667**
Expansion.Alt.Chosen alt.	0.9091	0.9091	1.0000	0.8000	–	–	–	–	–
Expansion.Conjunction	0.9651	0.9704	0.9503	0.9474	0.9628	0.9598	0.9650	**0.9652**	0.9585
Expansion.Exception	–	–	–	–	–	–	–	–	–
Expansion.Instantiation	1.0000	0.9474	1.0000	1.0000	**0.8000**	**0.8000**	**0.8000**	**0.8000**	**0.8000**
Expansion.Restatement	0.2500	0.5000	0.2222	0.6000	0.2857	0.	**0.5000**	0.4444	**0.5000**
Temporal.Async.Precedence	0.9592	0.9375	0.9211	0.9600	0.9351	0.8205	**0.9620**	0.9487	0.9487
Temporal.Async.Succession	0.8539	0.7816	0.8148	0.6863	0.8571	0.8288	0.8522	0.8319	**0.8739**
Temporal.Synchrony	0.8171	0.8250	0.7473	0.7066	**0.6885**	0.6055	0.6838	0.6545	0.6296
Overall	0.9059	0.9014	0.8959	0.8753	0.7680	0.7230	**0.7820**	0.7717	0.7766

Table 5.10

Confusion matrix for `Explicit` relations from the FR-en system on English blind dataset. Bold indicates counts that occur at least one-fourth of the time.

Predicted \ Truth	Comparison.Concession	Comparison.Contrast	Contingency.Cause.Reason	Contingency.Cause.Result	Contingency.Condition	Expansion.Conjunction	Temporal.Async.Precedence	Temporal.Async.Succession	Temporal.Synchrony	Other (6 sense labels)	Total	Precision
Comparison.Concession	4	–	–	–	–	–	–	–	–	–	4	1.0000
Comparison.Contrast	**64**	**23**	–	–	–	2	–	–	5	2	96	0.2396
Contingency.Cause.Reason	–	1	**22**	–	–	–	–	–	1	–	24	0.9167
Contingency.Cause.Result	–	–	–	**11**	–	–	1	–	1	1	14	0.7857
Contingency.Condition	3	–	–	–	**26**	–	–	–	–	–	29	0.8966
Expansion.Conjunction	3	2	–	1	–	**207**	1	1	1	4	220	0.9409
Temporal.Async.Precedence	–	–	–	–	–	–	**36**	1	–	–	37	0.9730
Temporal.Async.Succession	–	–	–	–	–	–	–	**48**	–	–	48	1.0000
Temporal.Synchrony	2	3	**9**	–	–	1	1	14	**42**	–	72	0.5833
Other (6 sense labels)	–	–	–	–	–	–	1	–	–	11	12	–
Total	76	29	31	12	26	210	40	64	50	18		
Recall	0.0526	0.7931	0.7097	0.9167	1.0000	0.9857	0.9000	0.7500	0.8400	–		

labels (EntRel, Expansion.Alt, Expansion.Alt.Chosen alt., Expansion.Exception, Expansion.Instantiation, Expansion.Restatement) occur less than 5 times, so we merged their corresponding rows and columns to make the confusion matrix clearer. A perfect recall-score and high precision-score is achieved by Contingency.Condition. Second highest recall-score and high precision-score is achieved by Expansion.Conjunction, which is the most common sense label in the train dataset (see Tab. 2.3) and contributes a lot to the overall performance. It is interesting to note, that Comparison.Concession is most of the time confused with Comparison.Contrast. A highest two precision-scores with a noticable amount of samples are achieved for Temporal.Async.Precedence and Temporal.Async.Succession, probably because discourse connectives for the temporal ordering of events have only one meaning.

5.3.2 *Analysis of non-`Explicit` relations*

We continue with an analysis of per-sense results in F_1-score in Tab. 5.11 and confusion matrix in Tab. 5.12 for non-`Explicit` relations on English datasets. Non-`Explicit` relation types are `Implicit`, `AltLex`, and `EntRel` (described in Section 2.1).

We analyse the *per-sense results in* F_1*-score for non-`Explicit` relations* primarily on the blind dataset, which is presented in Tab. 5.11. Predicting non-`Explicit` relations seems to be a substantially more difficult problem. All systems completely fail to recognize 6 sense labels, even on the valid and test datasets. This is not unusual for sense labels with only a few samples, but there should be enough training samples for sense labels Comparison.Contrast and Temporal.Async.Precedence. This suggests that the target concept for these sense labels is unsuitable for current methods, and thus much more research and a completely new approach is needed. The system by Mihaylov & Frank [19] substantially outperforms our models at Expansion.Conjunction and Expansion.Instantiation. The FR-en system achieves competitive performance on Contingency.Cause.Reason and Contingency.Cause.Result.

We also explore the *confusion matrix for non-`Explicit` relations* from the FR-en system on the blind dataset, which is presented in Tab. 5.12. Because the confusion matrices for both our systems (FR-en, FR-ench) look similar, we present only one of them. Seven sense labels (Contingency.Condition, Expansion.Alt, Expansion.Alt.Chosen alt., Expansion.Exception, Temporal.Async.Precedence, Temporal.Async.Succession, and Temporal.Synchrony) occur less than 5 times, therefore

	valid		test		blind					
Sense label	FR-en	FR-ench	FR-en	FR-ench	FR-en	FR-ench	[19]	[26]	[20]	simple-en
Comparison.Concession	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
Comparison.Contrast	0.0566	0.	0.0976	0.	0.0851	**0.0606**	0.	0.	0.	0.
Contingency.Cause.Reason	0.2699	0.3170	0.3010	0.2949	0.2523	0.2537	0.2136	0.2381	**0.3103**	0.2604
Contingency.Cause.Result	0.0984	0.1231	0.1441	0.0734	0.2128	0.1860	0.1818	0.0426	0.1818	**0.2385**
Contingency.Condition	–	–	–	–	–	–	–	–	–	–
EntRel	0.5646	0.5387	0.4691	0.4316	0.4790	0.5273	0.5424	0.5035	**0.5516**	0.4479
Expansion.Alt	–	–	–	–	0.	0.	0.	0.	0.	0.
Expansion.Alt.Chosen alt.	0.	0.	0.	0.	–	–	–	–	–	–
Expansion.Conjunction	0.3025	0.0305	0.2817	0.0497	0.2344	0.0323	**0.3444**	0.2385	0.2644	0.2320
Expansion.Exception	–	–	–	–	–	–	–	–	–	–
Expansion.Instantiation	0.2785	0.	0.2574	0.0274	0.1724	0.0465	**0.2807**	0.0426	0.2500	0.1538
Expansion.Restatement	0.2535	0.1515	0.2156	0.0965	0.2520	0.0859	0.1963	**0.3125**	0.2991	0.0619
Temporal.Async.Precedence	0.0714	0.	0.2000	0.	0.	0.	0.	0.	**0.1250**	0.
Temporal.Async.Succession	0.	0.	0.	0.	–	–	–	–	–	–
Temporal.Synchrony	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
Overall	0.3445	0.3378	0.2962	0.2636	0.3109	0.3216	0.3451	0.3384	**0.3767**	0.2680

Table 5.11 Per-sense results in F_1-score for non-`Explicit` relations on English datasets.

Table 5.12

Confusion matrix for non-Explicit relations from the FR-en system on English blind dataset. Bold indicates counts that occur at least one-fourth of the time.

Predicted / Truth	Comparison.Concession	Comparison.Contrast	Contingency.Cause.Reason	Contingency.Cause.Result	EntRel	Expansion.Conjunction	Expansion.Instantiation	Expansion.Restatement	Other (7 sense labels)	Total	Precision
Comparison.Concession	0	–	–	–	–	–	–	–	–	0	1.0000
Comparison.Contrast	2	2	–	–	–	–	–	–	–	21	0.0952
Contingency.Cause.Reason	5	2	**14**	4	5	13	5	19	2	69	0.2029
Contingency.Cause.Result	–	–	–	5	–	–	1	4	–	10	0.5000
EntRel	**11**	**10**	6	8	**114**	**45**	**15**	**60**	7	276	0.4130
Expansion.Conjunction	**10**	**8**	**13**	**11**	**49**	**32**	6	26	5	160	0.2000
Expansion.Instantiation	–	1	–	1	5	1	5	4	–	17	0.2941
Expansion.Restatement	2	3	8	5	23	17	8	31	3	100	0.3100
Other (7 sense labels)	–	–	–	–	–	–	–	–	0	0	–
Total	30	26	42	37	200	113	41	146	18		
Recall	0.	0.0769	0.3333	0.1351	0.5700	0.2832	0.1220	0.2123	–		

we merged their corresponding rows and columns to make the confusion matrix clearer. The highest recall-score is achieved by EntRel, which is the most common sense label and represents 23% of non-`Explicit` relations from the train dataset (see Tab. 2.3). The second most common is Expansion.Conjunction with 18%, but it performs much worse in terms of precision-score and recall-score. The high frequency of these two sense labels would explain why our method has a bias to confuse all sense labels with EntRel or Expansion.Conjunction. The off-diagonal counts for these two sense labels indicate that our method also has trouble distinguishing between them. This is not so surprising, because the main distinction between them is that in one case there is a coreferent entity mention and in the other mentioned entities are non-related. More interesting are the situations that Expansion.Restatement is almost half of the time confused with EntRel, and that Contingency.Cause.Result is almost one-third of the time confused with Expansion.Conjunction. We analyse this two particular cases in the case study in Section 5.3.3.

5.3.3 Case study of particular errors

We continue with a case study of two particular cases of errors for non-`Explicit` relations in Tab. 5.13 and Tab. 5.14 on English. To improve our understanding of the confused sense labels show a few examples of a sense label and the predicted probabilities by the FR-en system.

Although the sense label Expansion.Restatement represents 14.4% of the English train dataset, the recall-score for the FR-en system is very low (only 0.2123). The confusion matrix on the English blind dataset (see Tab. 5.12) indicates that it is correctly classified only 21% of the time, but 41% of the time it is confused with EntRel. In Tab. 5.13 we analyse this type of error by presenting a few examples of Expansion.Restatement with probabilities predicted by the FR-en system. Although all examples represent situations where `arg2` is a restatement or rephrasing of `arg1`, even non-expert human readers have difficulties in determining most cases. Let us examine the first example in Tab. 5.13 (marked with bold). To interpret it as Expansion.Restatement we must know that "woof" is a sound related to "dog" and "pendant" is hanging from a chain worn around the neck like as it was attached to a "collar". If we lack this prior knowledge about dogs and chains, we can only notice that both arguments are talking about the "pedometer", therefore they are connected by this entity, and should be labelled as EntRel. Because the FR-en system predicts EntRel with a much higher

Table 5.13

Examples of EXPANSION.RESTATEMENT with probabilities predicted by the FR-en system.

arg1	arg2	conn	RESTA.	ENTREL
The pedometer is called "Wandant", from Japanese "Wan", equivalent to "wool", and the "dant" of "pendant	The latter refers to the pedometer, because it is attached to a dog collar		0.0551	**0.5756**
Taal said in court that he knew Mam Sair Ceesay	I used to work at the Daily Observer as Managing Director up to November 2007. I know the accused," he said		0.0612	**0.7855**
Only the 1972 Dolphins finished the season with perfection	They played a 14-game schedule with three playoff games		0.1425	**0.2773**
You've never seen a Columbine done by a black child	Never		0.1505	**0.2700**
U.S. military commanders have decided to step up their counterinsurgency efforts in the city	They will add more troops and work toward reconstruction		**0.1658**	0.1625
he would request Prime Minister Rudd stay true to his promise that "no one would go without compensation"	I think the state Government should be our white knight		0.1672	**0.1678**
Whatever change that may take place in the world, our friendship with the African people will not change	He described this friendship as 'unbreakable		**0.1766**	0.1230
we will provide that	We will obviously give special thought to how we are going to do that and we will take into account the families' wishes		**0.1924**	0.1126
that his council would host a fitting tribute to Mr Brock	The council will make plans for a fitting tribute in honour of Peter's life and career in the coming days		0.2153	**0.2268**
Nigel Farage, leader of the UK Independence Party, questions the assertions that there is scientific consensus on global warming	At best, he said, there is uncertainty		**0.2385**	0.1898
that "our suspects here are the lawless MILF group because of the types of improvised explosive device that were used	The IEDs contained mortar rounds which are their signature		**0.2575**	0.1658
although 2004 saw sharp increases in young voter participation	That year, 47 percent of 18 to 24-year-olds voted, compared with 36 percent in the 2000 election		**0.2799**	0.0615
As soon as I saw it, there was no question about it	I looked at 211 fantastic sites all over Europe, but here it is - the dune size and the ocean front		**0.2859**	0.1504

All examples represent non-Explicit relations with sense label EXPANSION.RESTATEMENT (column RESTA.), but are sometimes misclassified as ENTREL. EXPANSION.RESTATEMENT is used when arg2 is a restatement or rephrasing of arg1. ENTREL is used when both arguments are connected only by the fact that they are about the same entity or person.

Table 5.14

Examples of CONTINGENCY.CAUSE.RESULT with probabilities predicted by the FR-en system.

arg1	arg2	conn	CAUSE.RES.	CONJ.
In the announcement, the Chairman of the Nokia Board of Directors stressed an expected shift of focus from hardware to software	Elop's strong software background and proven record in change management will be valuable assets		0.0669	**0.3605**
I used to work at the Daily Observer as Managing Director up to November 2007	I know the accused		0.0707	**0.2505**
I haven't been to town for about two years	Wander round the town and have a cup of tea... I'd love that		0.0820	**0.1869**
A further near miss was suffered by Bottas when he ran off-track after becoming caught in a battle between Räikkönen and Ricciardo	His team radioed to promise a complaint would be made to race director Charlie Whiting		0.0831	**0.2115**
to defer these taxes until economic conditions improve and James Hardie's profitability returns	In this way, the Australian government would allow James Hardie's taxes to go to the asbestos fund	In this way	0.0961	**0.6119**
he was involved in only one other lawsuit involving ORU, concerning its hospital building, and it was settled before trial	I hate to call him a liar, but he's a liar		0.1241	**0.1850**
but only if a Commons committee had decided the MP could be recalled	This latter requirement will make it "impossible to recall anyone" according to Goldsmith	This latter requirement will make it	**0.3167**	0.0613
The data are presented graphically on a custom website that makes trends in the dog's activities easy to understand at a glance	This helps owners get a stronger sense of their dog's health, while enabling communication with the dog	This helps	**0.5741**	0.0585
Alias" currently airs Thursdays at 8:00 PM EST on ABC	This has led to a drop in ratings as it is opposite CBS' popular reality TV show "Survivor: Guatemala	This has led	**0.8309**	0.0278
A train travelling from Tel Aviv to Haifa struck a truck at a junction	As a result of the impact, the first three carriages of the train were derailed	As a result of the impact	**0.9785**	0.0014

All examples represent non-Explicit relations with sense label CONTINGENCY.CAUSE.RESULT (column CAUSE.RES.), but are sometimes misclassified as EXPANSION.CONJUNCTION (column CONJ.). CONTINGENCY.CAUSE.RESULT is used when arg2 is a causal result or consequence of arg1. EXPANSION.CONJUNCTION is used for coordinating conjunctions joining phrases of equal rank or simultaneously occuring events.

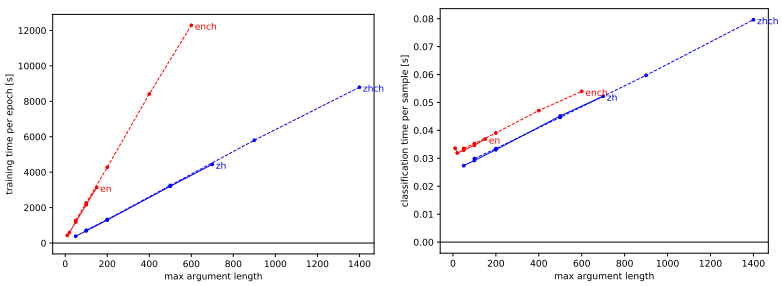
probability, we conclude that it lacks this specific prior knowledge.

Similarly, the sense label CONTINGENCY.CAUSE.RESULT represents 8.6% of the English train dataset, the recall-score for the FR-en system is even lower (only 0.1351). The confusion matrix on the English blind dataset (Tab. 5.12) indicates that it is correctly classified only 14% of the time, but 30% of the time it is confused with EXPANSION.CONJUNCTION. In Tab. 5.14 we analyse this type of error by presenting a few examples of CONTINGENCY.CAUSE.RESULT with probabilities predicted by the FR-en system. Although all examples represent situations where `arg2` is a causal result or consequence of `arg1`, non-expert human readers also have difficulties in determining misclassified cases. Let us examine the second example in Tab. 5.14 (marked with bold). To interpret it as CONTINGENCY.CAUSE.RESULT we must know that the "Managing Director" is being "accused" and that you know someone if you work with him. If we lack this information about the director and prior experience on human relations, we conclude that both arguments describe facts of equal rank and should therefore be labelled as EXPANSION.CONJUNCTION. Because the FR-en system predicts EXPANSION.CONJUNCTION with a much higher probability, we conclude that it again lacks this specific prior knowledge.

5.4 Time complexity

The proposed FR system is basically a system built around a machine learning algorithm for classification. An important characteristic of any machine learning algorithm is its training and classification time complexity.

In our case we use a neural network model described in Chapter 4 which is implemented by unrolling all LSTM layers to the maximal length of each text span. Consequently, the *theoretical time complexity* of our model is the same as for any FFNN with the equivalent number of connections. Because the number of computations necessary to train on or to classify any discourse relation stays the same, the time complexity in big-O asymptotic notation to process a single discourse relation is $O(1)$ and to process N discourse relations is $O(N)$. Due to this obviously linear nature, we will skip the empirical evaluation with respect to the number of discourse relations. On the other hand, the parameters that describe the maximal length of each text span and the number of focused downstream RNNs have a linear effect on the number of connections in the focused RNNs layer. The time complexity to process a single discourse relation with respect to the maximal length of text spans (M) is therefore $O(M)$. Similarly, the time complexity to process a single discourse relation with respect to the number of focused downstream RNNs (n) is $O(n)$.



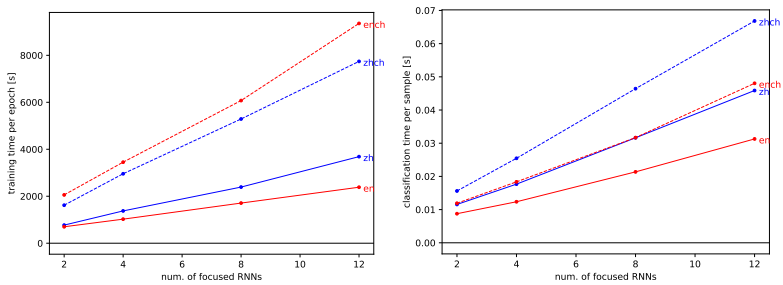
(a) Training time per epoch.

(b) Classification time per sample.

Figure 5.1

Measurement of training (a) and classification (b) time of the FR system with respect to the maximal length of `arg1` and `arg2` (M) on Chinese and English datasets at word level (FR-zh, FR-en) and character level (FR-zhch, FR-ench).

We performed an *empirical time complexity* analysis to confirm the theoretical results. In Fig. 5.1 we show the measured times by changing the maximal length of `arg1` (m = M) and `arg2` (m' = M). Up to an acceptable measuring error, the measurements confirm the theoretical time complexity of $O(M)$ for both training and classification. As



(a) Training time per epoch.

(b) Classification time per sample.

Figure 5.2

Measurement of training (a) and classification (b) time of the FR system with respect to the number of focused downstream RNNs (n) on Chinese and English datasets at word level (FR-zh, FR-en) and character level (FR-zhch, FR-ench).

expected, for a fixed M the times do not depend on the level of input representation, but only on the chosen language, which defines the number of focused downstream RNNs (n). In Fig. 5.2 we show the measured times by changing the number of focused downstream RNNs (n). Up to an acceptable measuring error, the measurements confirm the theoretical time complexity of $O(n)$ for both training and classification.

We ran the experiments on a single PC with a middle-range GPU from 2016 (Intel Core i7 3.20GHz with GeForce GTX 980 Ti). In all settings the total training time was extremely slow, but classification time incredibly fast. For the FR-en system, the training time per epoch was around 35 minutes, the total training time 9 hours, and the classification time per sample 0.030 seconds. For the FR-ench system, the training time per epoch was around 83 minutes, the total training time 105 hours, and the classification time per sample 0.042 seconds. For the FR-zh system, the training time per epoch was around 53 minutes, the total training time 17 hours, and the classification time per sample 0.045 seconds. For the FR-zhch system, the training time per epoch was around 96 minutes, the total training time 47 hours, and the classification time per sample 0.06 seconds. Due the sequential nature of performing empirical time complexity measurements on a single machine, unpredictable number of epochs, and some issues, performing these experiments took more than a month.

Note that it is impossible to compare our processing times with other systems, because their papers do not report this information. But from submissions of the CoNLL 2016 Shared Task, we can determine that our classification times are among the fastest.

6

Conclusion

In this chapter we briefly summarize the principal scientific contributions, present open challenges for future research, and where it leads to.

6.1 Scientific contributions

Our work described in this dissertation was presented at international conferences and published in renowned scientific journals. It was therefore internationally reviewed and discussed. Let us briefly summarize our contributions to science and their main features (also described in Section 1.3).

1. *Multi-dimensional RNN-attention mechanism* (focused RNNs).

   To outline the features of this contribution in comparison to related work at the time when the idea was first conceived:

   - a novel neural network layer with an attention mechanism for constructing sentence/argument embeddings
   - the first using multi-head or multi-dimensional attention weights, instead of attending to only a single aspect at a time
   - the first using RNNs for production of attention weights, instead of computing them as the inner product with a query vector
   - computes all attention weights in one pass, instead of recomputing them for different query vectors when focusing on different aspects
   - uses RNNs for aggregation of argument embeddings, instead of a sum of the weighted vectors

   This contribution is covered in Chapter 3. Its concept was introduced in Weiss & Bajec [22] and more details published in Weiss & Bajec [23].

2. *Language-independent method for sense classification* (FR system).

   To outline the features of this contribution in comparison to related work:

   - a novel method for sense classification of shallow discourse relations based on focused RNNs layer

- the first using a single end-to-end trainable model, instead of a complex pipeline of substantially different models to handle specific types and situations of discourse relations
- the first language-independent approach that requires no hand-engineered features or external resources, not even pre-trained word embeddings
- the first that can be applied at the word and character level inputs without any preprocessing
- provides a simple data augmentation technique to produce more samples, instead of training only on given samples

This contribution is covered in Chapter 4. Its predecessor, a more complex two-model system [22], received the first award for Chinese at CoNLL 2016 Shared Task by a large margin. We generalized upon it and published the FR system in Weiss & Bajec [23]. It achieves a new state-of-the-art performance on Chinese datasets.

6.2 *Future directions*

We managed to build a language-independent method for sense classification that requires only a training dataset to work. Because it learns only from labelled samples, a straightforward way of improving its performance would be to add more samples and increase the dataset size. One way is by only adding hand-picked informative samples for poorly-performing sense labels. Especially for many less-frequent sense labels that are the result of a highly imbalanced class distribution in all datasets (see Tab. 2.2 and Tab. 2.3). In order to also preserve the naturally-occurring class distribution in current datasets we would need to acquire a corpus of newspaper articles with a similar content and writing style as CDTB and PDTB. Multiple linguists would need to manually annotate it by following the same annotation scheme and rules. This is a highly time-consuming process, because a reasonable degree of inter-annotator agreement also needs to be achieved.

Instead of improving the performance, we could additionally confirm the language-independence of the FR system. It would make sense to apply it as such on even more languages, not just Chinese and English. Although annotated corpora exists for a few languages, such as Arabic, German, Italian, Czech, Hindi, or Turkish, most of them

only cover discourse connectives and `Explicit` relations, whereas `Implicit` relations are not yet part of the annotation. Most of them are much smaller then CDTB or PDTB and even the biggest one has less than 5700 discourse relations. As our method learns only from labelled samples, it is improbable that it would work in such situations. On the other hand, nothing in our method is limited to the scope of written newspaper articles or monologue, and we believe that it can be applied to shallow discourse relations on different kinds of text. The FR system could even be applied to different discourse structures, labels, and as part of more complex tasks, such as parsing of a dialogue or multi-party conversation.

We performed many experiments and tried to improve the FR system in different ways, but there are still some unexplored directions related to the inputs of our method. One could add new input representations that are between the currently supported word and character levels. English datasets contain only 87 different input characters (upper and lower letters, numbers, punctuation) that individually do not convey any meaning. One needs to read on average 4.96 characters to retrieve one of 43918 words and the meaning it carries. Subword units of different lengths, such as N-grams [58] which are N-character slices of longer strings, could be used to bring the input symbols closer to the meaning. A different approach is to combine a word level representation constructed from characters embeddings with pre-trained word embeddings [59]. In theory such embeddings of input symbols bring together the best of both worlds. Unfortunately, it is unclear how to technically implement embeddings at several levels simultaneously in the FR system. Furthermore, because we are learning everything from scratch only from labelled samples, any data augmentation or generation technique capable of producing valid labelled samples would improve its performance. To make the model more robust to noise and improve the learning of task-specific word or character embeddings, we already introduced a simple data augmentation technique which does not require external resources. A richer data augmentation can be performed with the help of a language-dependent thesaurus, like the WordNet lexical database. Because discourse relations do not change as long as the general meaning of the arguments stays the same, we could generate new samples by replacing words with words with nearly the same meaning (synonyms) or with a broader meaning (hypernyms). A more advanced approach for generating labelled samples would be to adapt a Generative Adversarial Network (GAN) [60]. A GAN is a system of two neural networks, called a generator and a discriminator, that contest with each other in an adversarial zero-sum game frame-

work. Theoretically we could set up the generator network to synthesize the text and sense label for fake shallow discourse relations, and the discriminator network to distinguish between real and fake samples. In an ideal scenario the generator network would learn to synthesize valid shallow discourse relations indistinguishable from real samples. However, convergence issues and difficulties when dealing with discrete data hinder the direct application of GAN to NLP tasks. Modified objectives, like TextGAN, address these issues, but large training times still present an open issue for more complex NLP tasks.

A different direction for improvements is connected with modifying the model in the FR system. We already attempted to fine-tune the model hyper-parameters for each setting as described in Section 4.3.2. It is interesting to note that most hyper-parameters did not substantially impact the performance and the same model hyper-parameters can be used almost out-of-the-box on all settings. This suggests that changes to the neural network architecture might be necessary to further improve its performance. Unfortunately, large training times and far too many ways how it can be changed make this infeasible even with genetic algorithms and advancements in the field of automated neural architecture search. On the other hand, we believe that adding hand-engineered features and external resources to our method would improve its performance. This is the exact opposite of what we try to accomplish in this dissertation, because it leads towards complex pipelines, dependence on language-specific resources, and designing specifically for a given language and task.

The whole task of text-level discourse parsing is still a difficult problem and far from being solved. We have demonstrated that sense classification of shallow discourse relations can be learned in an end-to-end manner. A challenging direction for future research would be to also build an end-to-end trainable model for the argument extraction task and combine it with the FR system into a complete system for shallow discourse parsing. The main problem is that it is not obvious how to model the overlapping discourse relations in a manner suitable for neural networks. In argument extraction we need to extract the related pieces of text (`arg1`, `arg2`, `conn`, `punc`), that not necessarily represent a contiguous piece of text and often overlap with other discourse relations. One would need to construct an end-to-end trainable model capable of expressing multiple overlapping sets in an unordered fashion. This can not be accomplished with simple approaches, like word tagging tasks, because individual words are often a part of multiple discourse relations.

Even in its current state it would be interesting to integrate the FR system as a component or additional source of features into a structure-enabled NLP application, such as statistical machine translation, text summarization, sentiment analysis, question generation, coherence modelling, and discourse parsing. Nevertheless, future linguistic research on different applications is still needed to improve our cognitive model of the discourse phenomena and discourse comprehension.

6.3 Concluding remarks

In this dissertation, we approach the most challenging part of text-level discourse parsing from the perspective of how a child learns through samples without explicit teaching. Instead of depending on hand-engineered features, external resources, and designing a system specifically for a given language and task, we pursue a language-independent approach for sense classification of shallow discourse relations. In Chapter 3 we first present our focused RNNs layer, a novel neural network layer with an attention mechanism for constructing sentence/argument embeddings. In Chapter 4 we use it in our FR system, a novel method for sense classification of shallow discourse relations based on focused RNNs layer.

The most important and unique characteristic of the FR system is that it can be used almost out-of-the-box on any language, set of sense labels, or level of input representation. This is possible, because it consists of only a single end-to-end trainable model, instead of a complex pipeline of substantially different models. We have confirmed its language-independence by successfully applying almost the same model hyper-parameters on two substantially different languages, but also by providing its input at the word and even character levels. It is true that the model still needs to be trained on labelled samples for each language, but it does not require any preprocessing, hand-engineered features or external resources, not even pre-trained word embeddings.

We compared the FR system with winning systems and strong baselines on Chinese and English using the official datasets and methodology of the CoNLL 2016 Shared Task [16]. It improved 8% (with 0.7477 F_1-score) over best overall results of other systems on the Chinese blind dataset, but did not fall (with 0.5170 F_1-score) a lot behind state-of-the-art on English blind dataset. Given that English is the most explored language with most advanced language technologies, we expected that systems carefully designed for it will substantially outperform our language-independent approach. How-

ever, we can not compare F_1-scores between languages due to huge differences in the size of datasets, grammar, encoding of words, structure of sentences, sense labels, and especially their distribution. All known systems for English have substantially lower performance scores than systems for Chinese. This indicates that automated sense classification on English is much more difficult than on Chinese. This difference is clearly observed for the FR system that can be applied on both languages. Because its performance depends on the language, we can not claim it is completely language-independent, but it is independent with respect to its inputs and architecture. This drop in performance when switching to another language could be mitigated by extending the datasets, adding language-specific features, external resources, or additional information. We first analysed its overall performance in terms of F_1-score and Brier-score, then in more detail with per-sense results and confusion matrices for `Explicit` and non-`Explicit` relations, and performed a case study of errors on English. We also analysed its training and classification time complexity. To qualitatively assess the contribution of some design choices we also performed an ablation study.

To conclude, we believe that automated discourse parsing and analysis, especially sense classification of `Implicit` relations, is a crucial next step in natural language understanding. Even though the theoretical grounds for this linguistic phenomena are not fully understood, our single neural network model is capable of learning the necessary concepts for sense classification without manual feature engineering and external resources. Furthermore, it is likely that larger amounts of training data or advanced data augmentation techniques would bring the performance of the FR system closer to a human level. We feel that such an approach is not only beneficial for automated sense classification of shallow discourse relations, but will inspire researchers to adapt it for more complex NLP tasks.

BIBLIOGRAPHY

[1] Guasti MT (2002) *Language acquisition: The growth of grammar* (MIT press, Cambridge, MA, USA).

[2] Webber B, Stone M, Joshi A & Knott A (2003) Anaphora and Discourse Structure. *Computational Linguistics*. DOI: 10.1162/089120103322753347.

[3] Webber B, Egg M & Kordoni V (2012) Discourse structure and language technology. *Natural Language Engineering*. DOI: 10.1017/S1351324911000337.

[4] Lin Z, Liu C, Ng HT & Kan MY (2012) Combining Coherence Models and Machine Translation Evaluation Metrics for Summarization Evaluation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, July (Association for Computational Linguistics, Jeju Island, Korea), pp. 1006–1014.

[5] Maslennikov M & Chua Ts (2007) A Multi-resolution Framework for Information Extraction from Free Text. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, June (Association for Computational Linguistics), pp. 592–599.

[6] Meyer T & Webber B (2013) Implicitation of Discourse Connectives in (Machine) Translation. In *Proceedings of the Workshop on Discourse in Machine Translation (DiscoMT)*. pp. 19–26.

[7] Wachsmuth H, Trenkmann M, Stein B & Engels G (2014) Modeling Review Argumentation for Robust Sentiment Analysis. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. pp. 553–564.

[8] Agarwal M, Shah R & Mannem P (2011) Automatic question generation using discourse cues. In *Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications* (Association for Computational Linguistics), pp. 1–9.

[9] Lin Z, Ng HT & Kan My (2011) Automatically evaluating text coherence using discourse relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1* (Association for Computational Linguistics), pp. 997–1006.

[10] Lin Z, Ng HT & Kan MY (2014) A PDTB-Styled End-to-End Discourse Parser. *Natural Language Engineering*. DOI: 10.1017/S1351324912000307.

[11] Mann WC & Thompson SA (1988) Rhetorical structure theory: Toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse*. DOI: 10.1515/text.1.1988.8.3.243.

[12] Prasad R et al. (2008) The Penn Discourse TreeBank 2.0. *Proceedings of the Sixth International Conference on Language Resources and Evaluation (LREC)*, pp. 2961–2968.

[13] Pitler E & Nenkova A (2009) Using Syntax to Disambiguate Explicit Discourse Connectives in Text. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, August (Association for Computational Linguistics, Suntec, Singapore), pp. 13–16.

[14] Pitler E, Louis A & Nenkova A (2009) Automatic Sense Prediction for Implicit Discourse Relations in Text. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. DOI: 10.3115/1690219.1690241.

[15] Xue N et al. (2015) The CoNLL-2015 Shared Task on Shallow Discourse Parsing. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning: Shared Task*. DOI: 10.18653/v1/K15-2001.

[16] Xue N et al. (2016) The CoNLL-2016 Shared Task on Multilingual Shallow Discourse Parsing. In *Proceedings of the CoNLL-16 Shared Task*. DOI: 10.18653/v1/K16-2001.

[17] Lin Z, Kan MY & Ng HT (2009) Recognizing Implicit Discourse Relations in the Penn Discourse Treebank. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. DOI: 10.3115/1699510.1699555.

[18] Wang J & Lan M (2015) A refined end-to-end discourse parser. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning: Shared Task*. DOI: 10.18653/v1/K15-2002.

[19] Mihaylov T & Frank A (2016) Discourse Relation Sense Classification Using Cross-argument Semantic Similarity Based on Word Embeddings. In *Proceedings of the CoNLL-16 Shared Task*. DOI: 10.18653/v1/K16-2014.

[20] Rutherford AT & Xue N (2016) Robust Non-Explicit Neural Discourse Parser in English and Chinese. In *Proceedings of the CoNLL-16 Shared Task*. DOI: 10.18653/v1/K16-2007.

[21] Wang J & Lan M (2016) Two End-to-end Shallow Discourse Parsers for English and Chinese in CoNLL-2016 Shared Task. In *Proceedings of the CoNLL-16 Shared Task*. DOI: 10.18653/v1/K16-2004.

[22] Weiss G & Bajec M (2016) Discourse Sense Classification from Scratch using Focused RNNs. In *Proceedings of the CoNLL-16 Shared Task*, edited by N Xue. DOI: 10.18653/v1/K16-2006.

[23] Weiss G & Bajec M (2018) Sense classification of shallow discourse relations with focused RNNs. *PLOS ONE*. DOI: 10.1371/journal.pone.0206057.

[24] Fraser B (1999) What are discourse markers? *Journal of pragmatics*. DOI: 10.1016/S0378-2166(98)00101-5.

[25] Zhou Y & Xue N (2012) PDTB-style Discourse Annotation of Chinese Text. *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, (July): 69–77.

[26] Oepen S et al. (2016) OPT : Oslo –Potsdam – Teesside Pipelining Rules , Rankers , and Classifier Ensembles for Shallow Discourse Parsing. In *Proceedings of the CoNLL-16 Shared Task*. DOI: 10.18653/v1/K16-2002.

[27] Kang X, Li H, Zhou L, Zhang J & Zong C (2016) An End-to-End Chinese Discourse Parser with Adaptation to Explicit and Non-explicit Relation Recognition. In *Proceedings of the CoNLL-16 Shared Task*. DOI: 10.18653/v1/K16-2003.

[28] Schenk N et al. (2016) Do We Really Need All Those Rich Linguistic Features ? A Neural Network-Based Approach to Implicit Sense Labeling. In *Proceedings of the CoNLL-16 Shared Task*. DOI: 10.18653/v1/K16-2005.

[29] Marcu D & Echihabi A (2002) An unsupervised approach to recognizing discourse relations. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. DOI: 10.3115/1073083.1073145.

[30] Rutherford A & Xue N (2015) Improving the Inference of Implicit Discourse Relations via Classifying Explicit Discourse Connectives. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (Association for Computational Linguistics, Denver, Colorado), pp. 799–808.

[31] Park J & Cardie C (2012) Improving Implicit Discourse Relation Recognition Through Feature Set Optimization. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, July (Association for Computational Linguistics), pp. 108–112.

[32] Rutherford AT & Xue N (2014) Discovering Implicit Discourse Relations Through Brown Cluster Pair Representation and Coreference Patterns. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics* (Association for Computational Linguistics, Gothenburg, Sweden), pp. 645–654.

[33] Chiarcos C & Schenk N (2015) A Minimalist Approach to Shallow Discourse Parsing and Implicit Relation Recognition. *Proceedings of the Nineteenth Conference on Computational Natural Language Learning - Shared Task*. DOI: 10.18653/v1/K15-2006.

[34] Ji Y & Eisenstein J (2015) One vector is not enough: Entity-augmented distributed semantics for discourse relations. *Transactions of the Association for Computational Linguistics*, 3: 329–344.

[35] Braud C & Denis P (2015) Comparing Word Representations for Implicit Discourse Relation Classification. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. DOI: 10.18653/v1/D15-1262.

[36] Liu Y, Li S, Zhang X & Sui Z (2016) Implicit Discourse Relation Classification via Multi-Task Neural Networks. In *Proceedings of the 30th Conference on Artificial Intelligence (AAAI 2016)*. pp. 2750–2756. arXiv: 1603.02776.

[37] Elman JL (1990) Finding structure in time* 1. *Cognitive science*. DOI: 10.1207/s15516709cog1402_1.

[38] Goller C & Küchler A (1996) Learning Task-Dependent Distributed Representations by Backpropagation Through Structure. In *IEEE International Conference on Neural Networks*. pp. 347–352.

[39] Hochreiter S & Schmidhuber J (1997) Long Short-Term Memory. *Neural Computation*. DOI: 10.1162/neco.1997.9.8.1735.

[40] Graves A & Schmidhuber J (2005) Framewise phoneme classification with bidirectional LSTM networks. *Proceedings of the International Joint Conference on Neural Networks*. DOI: 10.1109/IJCNN.2005.1556215.

[41] Mikolov T, Corrado G, Chen K & Dean J (2013) Efficient Estimation of Word Representations in Vector Space. In *Proceedings of the International Conference on Learning Representations (ICLR)*. pp. 1–12. arXiv: 1301.3781v3.

[42] Pennington J, Socher R & Manning CD (2014) GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. DOI: 10.3115/v1/D14-1162.

[43] Xu K et al. (2015) Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. In *Proceedings of the 32nd International Conference on Machine Learning*. pp. 2048–2057. arXiv: 1502.03044v3.

[44] Bahdanau D, Cho K & Bengio Y (2014) Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:14090473*, pp. 1–15. arXiv: 1409.0473.

[45] Vinyals O et al. (2015) Grammar as a Foreign Language. In *Advances in Neural Information Processing Systems*. pp. 2773–2781. arXiv: 1412.7449v3.

[46] Hermann KM et al. (2015) Teaching Machines to Read and Comprehend. In *Advances in Neural Information Processing Systems*. pp. 1693–1701. arXiv: 1506.03340v1.

[47] Sukhbaatar S, Szlam A, Weston J & Fergus R (2015) End-To-End Memory Networks. In *Advances in neural information processing systems*. pp. 2440–2448. arXiv: 1503.08895.

[48] Parikh AP, Täckström O, Das D & Uszkoreit J (2016) A Decomposable Attention Model for Natural Language Inference. *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP-16)*. DOI: 10.18653/v1/D16-1244.

[49] Vaswani A et al. (2017) Attention Is All You Need. In *Advances in Neural Information Processing Systems*. pp. 5998–6008. arXiv: 1706.03762.

[50] Liu Q, Zhang H, Zeng Y, Huang Z & Wu Z (2018) Content Attention Model for Aspect Based Sentiment Analysis. In *Proceedings of the 2018 World Wide Web Conference*. DOI: 10.1145/3178876.3186001.

[51] Collobert R et al. (2011) Natural Language Processing (almost) from Scratch. *Journal of Machine Learning Research*, 12: 2493–2537.

[52] Jin X et al. (2016) Deep Learning with S-shaped Rectified Linear Activation Units. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16 (AAAI Press, Phoenix, Arizona), pp. 1737–1743. arXiv: 1512.07030.

[53] Chollet F et al. (2015) Keras. URL: http://keras.io/.

[54] Kingma DP & Ba JL (2015) Adam: a Method for Stochastic Optimization. *International Conference on Learning Representations 2015*, pp. 1–15. arXiv: 1412.6980.

[55] Glorot X & Bengio Y (2010) Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of Machine Learning Research*, vol. 9. pp. 249–256.

[56] Srivastava N, Hinton G, Krizhevsky A, Sutskever I & Salakhutdinov R (2014) Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 15: 1929–1958.

[57] Cho K et al. (2014) Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. DOI: 10.3115/v1/D14-1179.

[58] Cavnar WB & Trenkle JM (1994) N-gram-based text categorization. In *Proceedings of SDAIR-94, 3rd annual symposium on document analysis and information retrieval*, vol. 161175.

[59] Lample G, Ballesteros M, Subramanian S, Kawakami K & Dyer C (2016) Neural Architectures for Named Entity Recognition. In *Proceedings of NAACL-HLT 2016*. DOI: 10.18653/v1/N16-1030.

[60] Goodfellow I et al. (2014) Generative Adversarial Nets. In *Advances in Neural Information Processing Systems 27*, edited by Z Ghahramani, M Welling, C Cortes, ND Lawrence & KQ Weinberger (Curran Associates, Inc.), pp. 2672–2680.

ABBREVIATIONS

NLP	natural language processing
PDTB	English Penn Discourse TreeBank 2.0
CDTB	Chinese Discourse Treebank 0.5
FFNN	feed-forward neural network
CNN	convolutional neural network
RNN	recurrent neural network
LSTM	long short-term memory
focused RNNs	our multi-dimensional RNN-attention mechanism
FR system	our method for sense classification of shallow discourse relations
FR-zh	FR system for Chinese using word level representations
FR-zhch	FR system for Chinese using character level representations
FR-en	FR system for English using word level representations
FR-ench	FR system for English using character level representations
random-zh	minimal baseline model for Chinese that returns sense labels uniformly at random
random-en	minimal baseline model for English that returns sense labels uniformly at random
majority-zh	minimal baseline model for Chinese that returns the most common sense label
majority-en	minimal baseline model for English that returns the most common sense label
simple-zh	strong baseline model for Chinese similar to FR system with simple LSTMs
simple-en	strong baseline model for English similar to FR system with simple LSTMs
en	English language
zh	Chinese language
train	Training dataset (for model fitting)
valid	Validation or development dataset (for tuning model hyper-parameters)

test	Test dataset from the same corpus as train dataset
blind	Blind test dataset from a slightly out-of-domain corpus
arg1	argument 1 of a shallow discourse relation
arg2	argument 2 of a shallow discourse relation
conn	connective of a shallow discourse relation
punc	punctuation of a shallow discourse relation

Učenje besedilnega razčlenjevanja diskurzov

Gregor Weiss

doktorska disertacija
predana
Fakulteti za računalništvo in informatiko
kot del izpolnjevanja pogojev za pridobitev naziva
doktor znanosti
s področja
računalništva in informatike

razširjeni povzetek

Ljubljana, 2019

Uvod

Motivacija Lahko bi rekli, da je naša sposobnost simboličnega razmišljanja in komunikacije ena glavnih človeških lastnosti. Vsak zdrav človeški otrok se brez težav nauči jezik brez eksplicitnega učenja, na podlagi pozitivnih primerov in na *jezikovno-neodvisni* način [1]. To se zgodi v odsotnosti eksplicitno podane strukture stavkov, slovničnih pravil in leksikonov besed, opisov njihovega pomena ter relacij med njimi. Pri tem ne potrebujejo skupka *ročno-izdelanih značilk* ali vzorcev, na katere naj bodo pozorni, še manj pa korakov cevovodnega procesiranja, prirejenih za vsak tip pogovora ali nalogo. Vse potrebno se naučijo iz primerov. Po drugi strani pa so računski pristopi za večino nalog procesiranja naravnega jezika (NLP) odvisni od vsega naštetega in so narejeni za specifičen jezik in nalogo. Takšnih sistemov ni mogoče celostno učiti od začetka-do-kraja ali enostavno prilagoditi novim primerom. Te omejitve so še bolj opazne pri ne-angleških jezikih, kjer napredek v NLP še zaostaja. Trdimo, da mora obstajati druga pot, da se lahko tudi računske metode naučijo kompleksnih NLP nalog v poljubnem jeziku samo na podlagi primerkov, podobno kot otroci.

Poljubno besedilo v naravnem jeziku pomeni več kot le vsota njegovih delov ali stavkov [2]. Da lahko resnično razumemo besedilo, moramo identificirati njegove dele in izluščiti dodatne semantične relacije, imenovane *diskurzne relacije* (angl. discourse relations) ali koherentne relacije. Čeprav so diskurzne relacije pomemben del vsakega jezika in pomembne za večje število aplikacij za NLP [3], še vedno ne razumemo tega pojava popolnoma.

Tekom zadnjih treh desetletij so lingvisti predlagali številne teorije diskurza za analizo jezika ne samo v notranjosti povedi, temveč tudi na nivoju stavkov in besedil [2, 11]. V pričujoči disertaciji smo izbrali teorijo *plitkih diskurznih relacij* (angl. shallow discourse relations), tudi imenovanih PDTB-stil, ker predstavljajo teoretično-nevtralni pristop in nudijo največji označen korpus [12]. Le-te definiriajo *diskurzne argumente* (`arg1`, `arg2`) kot dele besedila, ki predstavljajo specifičen pomen (abstraktni objekti, dogodki, stanja, dejstva in predlogi). In *diskurzno relacijo* kot semantično relacijo, ki opisuje na kakšen način je par diskurznih argumentov povezan med seboj in kakšen pomen ali *oznako smisla* lahko izluščimo iz njega. Nekatere diskurzne relacije so eksplicitno izražene z uporabo veznikov (`conn`, e.g. *while*, *but*, *unless*) ali nakazane z ločili (`punc`), vendar jih najpogosteje identificira šele bralec, ki želi razumeti besedilo. V poglavju 2.1 lahko najdete več podrobnosti o tem. Za ilustracijo si oglejmo nekaj primerov:

1. *[Jane fell over]*$_{arg1}$, *while [Tarzan helped her]*$_{arg2}$.

2. *[I want to go to China]*$_{arg1}$, *but [I prefer clean air]*$_{arg2}$.

3. 除非 *[* 火车 晚点 *]*$_{arg1}$ ， 否则 *[* 我 会在 九点 钟 到 那里 *]*$_{arg2}$ 。
(*Unless [the train is late]*$_{arg1}$ *, (otherwise) [I will be there at nine o'clock]*$_{arg2}$.)

V zgornjih primerih vsaka diskurzna relacijo vsebuje eksplicitno podan diskurzni veznik, zato jih imenujemo `Explicit`-ne relacije. Ljudem je enostavno identificirati diskurzni veznik in pomen ali oznako smisla diskurzne relacije, ki jo nakazuje. V primeru 1. *while* nakazuje časovno sosledje dveh dogodkov, v primeru 2. *but* nakazuje kontrast in v primeru 3. *unless* nakazuje alternativni izzid. Iz računalniškega vidika je dokaj jasno kako napovedovati takšne oznake smisla s skrbno izdelavo produkcijskih pravil, ki znajo razbrati funkcijo veznika [13].

1. *[Jane fell over]*$_{arg1}$. *[Tarzan helped her]*$_{arg2}$.

2. *[I want to go to China]*$_{arg1}$. *[I prefer clean air]*$_{arg2}$.

3. *[* 火车 晚点 *]*$_{arg1}$ 。 *[* 我 会在 九点 钟 到 那里 *]*$_{arg2}$ 。
(*[The train is late]*$_{arg1}$. *[I will be there at nine o'clock]*$_{arg2}$.)

Razmislimo, kako drastično se zgornje diskurzne relacije in njihov pomen spremeni, če odstranimo podčrtane diskurzne veznike. Z nekaj truda lahko izluščimo nove manjkajoče veznike in z njimi nov pomen ali oznako smisla vsake ne-`Explicit`-ne relacije. Primera 1. in 3. sedaj predstavljata vzrok in njegov rezultat (kakor da bi bil *so* prisoten), medtem ko primer 2. našteva osebne preference v konjunkciji (kakor da bi bil *and* prisoten). V besedilih z naravnim jezikom je zelo pogosto, da diskurzne relacije niso nakazane z diskurznimi vezniki. Takšni primeri so lahko težavni tudi za ljudi, saj je o oznaki smisla potrebno sklepati iz semantičnega konteksta, koherence argumentov, znanjem o svetu ali z drugimi sredstvi [14]. Iz računalniškega vidika so takšne situacije veliko bolj zahtevne in predstavljajo ozka grla celotnih sistemov.

Razčlenjevanje diskurza je naloga ekstrakcije diskurznih relacij. Sestavljena je iz lociranja `conn` in `punc`, ekstrakcije `arg1` in `arg2` in *klasifikacije smisla*, i.e. določiti kateri pomen ali oznako smisla lahko izluščimo. Izkaže se, da je izdelava avtomatiziranega razčlenjevalnika diskurza na besedilnem nivoju izjemno težavna. Še posebej, ker razlike med

oznakami smisla zahtevajo precizno semantično sklepanje, ki pa ga ni možno enostavno opisati s tradicionalnimi značilkami za NLP. Za izboljšanje stanja sta dve konferenci CoNLL 2015 in 2016 organizirali izziv [15, 16], osredotočen na razčlenjevanje diskurza in klasifikacijo smisla v plitkih diskurznih relacijah na angleškem in kitajskem jeziku. Tekom obeh let je bila klasifikacija smisla implementirana v 40 konkurenčnih sistemih za angleščino in 10 za kitajščino.

V splošnem se je izkazalo, da ne-`Explicit`-ne diskurzne relacije še vedno predstavljajo najzahtevnejši problem za razne aplikacije. Obstoječi sistemi za klasifikacijo smisla uporabljajo kompleksen cevovod sestavljen iz bistveno različnih modelov za obravnavanje specifičnih tipov in situacij v diskurznih relacijah. Ti modeli zahtevajo pred-procesiranje, ročno-izdelane značilke, zunanje vire in obsežno ročno nastavljanje za vsak jezik in množico oznak smisla.

Motivirani z načinom, kako se otroci naučijo jezik, smo se usmerili stran od slabosti in kompleksnosti obstoječih sistemov za klasifikacijo smisla. Poskušali smo izdelati jezikovno-neodvisno metodo za nalogo klasifikacije smisla v plitkih diskurznih relacijah.

Definicija problema V pričujoči disertaciji se osredotočamo na nalogo *klasifikacije smisla* v plitkih diskurznih relacijah, tako kot je opisana v okviru CoNLL 2016 Shared Task [16], in skušamo k njej pristopiti iz *jezikovno-neodvisne* smeri.

> *Definicija 1: Klasifikacija smisla* v plitkih diskurznih relacijah
> *Oznaka smisla* (angl. sense label) ali semantični razred opisuje pomen, ki ga lahko interpretiramo za diskurzno relacijo (e.g. contrast, causation, conjunction). Zaradi razlik med jeziki imamo za kitajščino definirano množico 10 oznak smisla (za popolni seznam glej tabelo 2.2) in za angleščino 21 oznak smisla (za popolni seznam glej tabelo 2.3).
> Na podlagi podanih dveh diskurznih argumentov (`arg1`, `arg2`), neobveznega veznika (`conn`) in neobveznih ločil (`punc`), je naša naloga napovedati *oznako smisla* diskurzne relacijo, ki jo le-ti predstavljajo.

Več podrobnosti o plitkih diskurznih relacijah lahko najdete v poglavju 2.1.

Definicija 2: Jezikovno-neodvisni pristop k problemom iz NLP:
Menimo, da je pristop *neodvisen od jezika*, če ni bil zasnovan posebej za določen jezik in ne zahteva nobenega pred-procesiranja, ročno-izdelanih značilk, zunanjih virov ali obsežnega finega prilagajanja za vsak jezik. Z drugimi besedami povedano, da je neodvisen od jezika glede na njegove vhode in arhitekturo ter ga lahko kot takšnega uporabimo na zelo različnih jezikih.

V poglavju 4 predstavljamo kako smo se lotili zastavljenega cilja v našem pristopu FR system, novi metodi za klasifikacijo smisla v plitkih diskurznih relacijah, ki temelji na focused RNNs plasti. S skoraj identičnimi hiper-parametri smo metodo uspešno uporabili na dveh bistveno različnih jezikih, angleščini in kitajščini (brez poznavanja kitajščine).

Prispevki k znanosti V luči naše motivacije in pomembnosti klasifikacije smisla v različnih aplikacijah predstavljamo v pričujoči disertaciji sledeče prispevke k znanosti:

1. *Več-dimenzionalni RNN-pozornostni mehanizem* (focused RNNs).

   Predstavljamo osredotočene rekurentne nevronske mreže (focused RNNs), nov tip plasti za nevronske mreže s pozornostnim-mehanizmom za izdelavo vložitev stavkov/argumentov. Njihov namen je preslikati argumente diskurznih relacij v več vektorskih prostorov, ki zakodirajo različne vidike vhodnih delov besedila. V času, ko smo si prvič zamislili focused RNNs plast (v začetku 2016), so obstajali samo eno-pozornostni mehanizmi, ki združujejo z obteženim povprečjem. Po našem najboljšem poznavanju je naš pristop prvi predstavil dva nova koncepta in se še vedno bistveno razlikuje od ostalih pozornostnih mehanizmov. Prvič, je prvi pozornostni mehanizem z več-glavnimi oziroma več-dimenzionalnimi utežmi za pozornost, namesto da bi se osredotočal samo na en vidik naenkrat. Drugič, je prvi pozornostni mehanizem, ki uporablja RNNje za pripravo uteži za pozornost, namesto da bi jih izračunal kot notranji produkt s poizvedovalnim vektorjem. Tretjič, z izračunom vseh uteži za pozornost v enem obhodu, namesto bi jih ponovno preračunaval za različne poizvedovalne vektorje med osredotočanjem na različne vidike. Četrtič, z uporabo RNNjev za združevanje argumentnih vložitev, namesto da bi le seštel obtežene vektorje. Naša focused RNNs plast je sestavljena iz filtrirnega RNN kateri sledi pomnoževalni filtrirni/usmerjevalni mehanizem, ki omogoča sledečim RNNjem, da se osredotočijo na različne vidike vhodnega zapo-

redja in ga projecirajo v več vložitvenih podprostorov. Te argumentne vložitve se lahko nato uporabijo za različne NLP naloge, kot je klasifikacija smisla.

Ta prispevek smo zajeli v poglavju 3. Koncept smo prvič predstavili v Weiss & Bajec [22] in objavili več podrobnosti v Weiss & Bajec [23].

2. *Jezikovno-neodvisna metoda za klasifikacijo smisla* (FR system).

   Predstavljamo novo metodo za klasifikacijo smisla v plitkih diskurznih relacijah, ki temelji na focused RNNs plasti, zato jo imenujemo *FR system*. Po našem najboljšem poznavanju naša metoda predstavlja unikaten pristop h klasifikaciji smisla, ki se razlikuje od obstoječih metod v mnogih pogledih. Prvič, je prva sestavljena samo iz enega modela, ki ga je možno učiti od začetka-do-kraja, namesto kompleksnega cevovoda sestavljenega iz bistveno različnih modelov za obravnavanje specifičnih tipov in situacij v diskurznih relacijah (brez razlik med `Explicit`-nimi in ostalimi tipi relacij, relacij v notranjosti stavkov ali med stavki, vrstnega reda argumentov). Drugič, je prvi jezikovno-neodvisni pristop, ki ne potrebuje nobenih ročno-izdelanih značilk ali zunanjih virov, niti pred-naučenih besednih vložitev. Za delovanje potrebuje samo učno množico, kar naredi metodo uporabno skorajda brez sprememb na kateremkoli jeziku in oznakah smisla. Tretjič, je prva metoda, ki se lahko uporablja tako na ravni besed kot na ravni znakov brez vsakršnega pred-procesiranja. Četrtič, predstavi preprost mehanizem bogatenja podatkov za proizvodnjo primerkov, namesto da bi se učila samo na podanih primerih. Našo metodo smo ovrednotili na uradnih podatkovnih zbirkah in po metodologiji izziva CoNLL 2016 Shared Task. Ne zaostaja veliko za najuspešnejšimi sistemi na angleškem jeziku, vendar presega ostale sisteme brez focused RNNs plasti za 8% na kitajski podatkovni zbirki. Najprej smo analizirali njegovo splošno uspešnost z F_1-oceno in Brier-oceno, nato podrobnosti z rezultati po posameznih oznakah smisla in matriko zamenjav za `Explicit`-ne in ne-`Explicit`-ne relacije ter izvedli študijo primerov napak na angleščini. Analizirali smo tudi časovno kompleksnost pri učenju in klasifikaciji. Izvedli smo tudi študijo izključitev, da smo lahko kvalitativno ocenili doprinos nekaterih načrtovalskih odločitev.

   Ta prispevek smo zajeli v poglavju 4. Naš prejšnji bolj kompleksni dvo-modelni sistem [22] je prejel prvo nagrado z visoko prednostjo na kitajski podatkovni zbirki na CoNLL 2016 Shared Task. Nato smo koncept posplošili in objavili FR system v Weiss & Bajec [23].

Več-dimenzionalni RNN-pozornostni mehanizem (focused RNNs)

Vložitve (angl. embeddings) predstavljajo preslikavo diskretnih objektov, ki nimajo naravne vektorske predstavitve (kot so besede ali stavki), v goste vektorje z realnimi vrednostmi. Znano je, da se nevronske mreže najboljše učijo nad gostimi vektorji, pri katerih posamezne dimenzije običajno nimajo ločenega pomena in vse vrednosti kot celota opisujejo dani objekt. Za NLP klasifikacijske naloge in klasifikacijo smisla je pogost pristop z nevronskimi mrežami, da se najprej preslika besede v pred-naučene besedne vložitve, nato zakodira stavke ali dele besedila v vektorske predstavitve fiksne dolžine prilagojene nalogi, imenovane *stavčne/argumentne vložitve* (angl. sentence/argument embeddings), in potem uporabi usmerjeno nevronsko mrežo (FFNN) za klasifikacijo. Največje razlike med pristopi so običajno v tem, kako se ustvarijo stavčne vložitve. V našem primeru jih imenujemo argumentne vložitve in pri njih je ključno, koliko semantične podobnosti in koherenčnih informacij, povezanih z diskurznimi relacijami, lahko zajamejo. Naš pristop spada med *nevronske pozornostne mehanizme* (angl. neural attention mechanisms), ki omogočajo, da lahko model avtomatsko usmerja pozornost na dele vhoda, ki so najbolj relevantni v vsakem koraku procesiranja, in prilagaja pozornost skozi čas.

Za izdelavo boljših vložitev smo si zamislili nov tip plasti za nevronske mreže, ki ga imenujemo osredotočene rekurentne nevronske mreže (focused RNNs). Le-ta predstavlja prvi več-dimenzionalni RNNpozornostni mehanizem za izdelavo vložitev stavkov/argumentov. Sestavljen je iz *filtrirnega RNN* (RNN^{filter}), pomnoževalnega *filtrirnega/usmerjevalnega mehanizma* in *sledečih osredotočenih RNNjev* (RNN_i).

V poglavju 3 slika 3.1 prikazuje diagram procesiranja v naši focused RNNs plasti z n sledečimi osredotočenimi RNNji. Vhod je lahko katerokoli zaporedje gostih vektorjev, ki jih želimo preslikati v več vektorskih podprostorov. Za klasifikacijo smisla je to običajno zaporedje besednih vložitev $x = [x^{(1)}, x^{(2)}, ..., x^{(m)}]$, ki predstavljajo stavek ali argument diskurzne relacije dolžine m, ki ga želimo zakodirati v argumentno vložitev y. Najprej filtrirni RNN (RNN^{filter}) deluje kot več-dimenzionalna primerjalna funkcija, ki za vsako besedno vložitev ustvari vektor uteži pozornosti $f^{(t)}$. Le-te se lahko interpretira kot relativno pomembnost vsake besedne vložitve pri združevanju različnih vidikov vhodnega zaporedja. Teoretično se za filtrirni RNN lahko uporabi poljubna vrsta RNNjev, a dvosmerna plast z dolgim-kratkoročnim spominom (LSTM) [40] s σ aktivacijsko funkcijo (enačba 3.1) se je izkazala nekoliko boljša. Filtrirni/usmerjevalni mehanizem nato pomnoži vsako utež $f_i^{(t)}$ z istoležno besedno vložitvijo tako, da ustvari obteženo be-

sedno vložitev $a_i^{(t)}$ namenjeno enemu sledečemu osredotočenemu RNNju. Na ta način uravnava, koliko vhodnega signala se bo posredovalo posameznim RNNjem. Tako pripravljeno obteženo vhodno zaporedje omogoča sledečim osredotočenim RNNjem (RNN_i), da se specializirajo ali osredotočijo na različne vidike vhodnega zaporedja. Vsak osredotočen RNN deluje kot agregacijska funkcija, ki v svojem notranjem stanju sestavlja vektorsko predstavitev fiksne dolžine b_i. Le-ta predstavlja projekcijo vhodnega zaporedja v podprostor argumentnih vložitev. Na koncu vse pridelane vektorje b_i konkateniramo/sestavimo v daljši vektor y, ki predstavlja stavčno/argumentno vložitev in se lahko uporablja pri različnih NLP nalogah, kot je klasifikacija smisla.

Omeniti bi bilo potrebno, da se koncept focused RNNs plasti bistveno razlikuje od ostalih nevronskih pozornostnih mehanizmov. Po našem najboljšem poznavanju predstavlja naš pristop prvi več-dimenzionalni RNNpozornostni mehanizem.

Več podrobnosti lahko najdete v poglavju 3. Sicer pa smo koncept prvič predstavili v Weiss & Bajec [22] in objavili več podrobnosti v Weiss & Bajec [23].

Jezikovno-neodvisna metoda za klasifikacijo smisla (FR system)

FR system je naša predlagana rešitev/metoda za klasifikacijo smisla ali pomena v plitkih diskurznih relacijah. Sestavljena je iz samo enega modela, ki ga je mogoče celostno učiti od začetka-do-kraja, iz koraka za pripravo vhodnih podatkov in postopka učenja, ki vključuje preprost mehanizem bogatenja podatkov med učenjem. Naš model neposredno sledi definiciji naloge klasifikacije smisla in obravnava vse vrste in smisle diskurznih relacij. Tudi ne potrebuje nobenih ročno-izdelanih značilk ali zunanjih virov, zaradi česar je jezikovno-neodvisen glede na vhodne podatke in arhitekturo. Je tudi prvi sistem za klasifikacijo smisla, ki se lahko uporablja tako na ravni besed kot na ravni znakov. Ker se metoda uči besedne vložitve prilagojene na nalogo od začetka, mi zgolj uvedemo preprost mehanizem bogatenja podatkov med učenjem. Prav tako je model odvedljiv od začetka-do-kraja, zato se ga lahko uči z vzvratnim razširjanjem napake (angl. backpropagation) na označenih primerih.

V poglavju 4 slika 4.1 prikazuje arhitekturo nevronske mreže našega FR system za klasifikacijo smisla, ki temelji na focused RNNs plasti. Vhod za vsako diskurzno relacijo je podan v obliki štirih odsekov besedila (angl. text span) v surovi obliki: za dva argumenta (`arg1`, `arg2`), opcijski veznik (`conn`) in opcijska ločila (`punc`). V duhu učenja od začetka-do-kraja ne izvajamo nobenega pred-procesiranja in delamo neposredno z odseki besedila predstavljenimi kot zaporedje vhodnih simbolov na nivoju besed ali na

nivoju znakov ($w = [w^{(1)}, w^{(2)}, \ldots, w^{(m)}$). Zaradi konsistentnosti procesiramo vse odseke besedila na popolnoma enak način, i.e. sledimo istim enačbam. Kjer je potrebno ločimo spremenljivke z apostrofi (e.g. w za `arg1`, w' za `arg2`, w'' za `conn`, and w''' za `punc`). Vsak odsek besedila je procesiran neodvisno od ostalih od začetka do konca, kjer je (t) trenutna pozicija po časovni dimenziji (e.g. t-ta beseda v odseku besedila). Najprej se plast za besedne/znakovne vložitve (angl. word/char embedding layer) nauči preslikovati vhodne simbole v vektorske predstavitve prilagojene nalogi, imenovane besedne ali znakovne vložitve $x^{(t)}$. Pri tem ne uporablja pred-naučenih besednih vložitev in začne z naključno inicializiranimi vektorji. Nato je vsako zaporedje besednih vložitev ($x = [x^{(1)}, x^{(2)}, \ldots, x^{(m)}$) neodvisno procesira naša focused RNNs plast. Plast focused RNNs je sestavljena iz filtrirnega RNN, pomnoževalnega filtrirnega/usmerjevalnega mehanizma in več sledečih osredotočenih RNNov. Le-ti projecirajo vsak odsek besedila v vektorsko predstavitev fiksne dolžine, ki jih imenujemo argumentna vložitev y. Nato konkateniramo argumentne vložitve vseh odsekov besedila (y za `arg1`, y' za `arg2`, y'' za `conn`, and y''' za `punc`) v daljše vektorje c in jih posredujemo dvo-plastni usmerjeni nevronski mreži (FFNN). Njen namen je napovedovanje verjetnosti p za oznake smisla. Na koncu FR system vrne oznako smisla z najvišjo verjetnostjo p kot rezultat klasifikacije smisla za dano plitko diskurzno relacijo.

Z uporabo istih plasti (z istimi utežmi) na vseh štirih odsekih besedila, vzpodbujamo, da so istoležne vložitve predstavljene v istem vektorskem prostoru, in preprečujemo prekomerno prileganje posebnostim posameznega odseka besedila.

Isto arhitekturo nevronske mreže smo uspešno uporabili na kitajski in angleški podatkovni zbirki, tako na nivoju besed (FR-zh, FR-en) kot na nivoju znakov (FR-zhch, FR-ench). Zaradi razlik med jeziki in razlik v oznakah smisla je bilo potrebno prilagoditi nekaj osnovnih parametrov (glej tabelo 4.1).

Več podrobnosti lahko najdete v poglavju 4. FR system smo v celoti objavili v Weiss & Bajec [23].

Evaluacija

Evaluacijo našega FR system smo izvedli na kitajščini v poglavju 5.2 (s sistemi FR-zh, FR-zhch) in na angleščini v poglavju 5.3 (s sistemi FR-en, FR-ench).

Pri tem smo primarno sledili uradni formulaciji naloge, podatkovnim zbirkam in metodologiji izziva CoNLL 2016 Shared Task [16], ki je bil organiziran v okviru konference CoNLL. Uspešnost našega FR system smo primerjali z zmagovalnimi sistemi in močni-

mi izhodiščnimi modeli. Najprej smo analizirali njegovo splošno uspešnost z F_1-oceno in Brier-oceno, nato podrobnosti z rezultati po posameznih oznakah smisla in matriko zamenjav za `Explicit`-ne in ne-`Explicit`-ne relacije ter izvedli študijo primerov napak na angleščini. Analizirali smo tudi časovno kompleksnost pri učenju in klasifikaciji. Izvedli smo tudi študijo izključitev, da smo lahko kvalitativno ocenili doprinos nekaterih načrtovalskih odločitev.

Na kitajski podatkovni zbirki v tabeli 5.1 oba naša sistema (FR-zh, FR-zhch) presežeta vse ostale sisteme na blind podatkovni zbirki. Sistem FR-zhch celo preseže naš prejšnji dvo-modelni sistem [22] za 2.5% na vseh relacijah. V primerjavi z ostalimi sistemi, ki ne uporabljajo focused RNNs plasti, pa preseže celo za več kot 8%. Pričakovali bi, da se bolj kompleksni in ročno nastavljeni sistemi odrežejo boljše od sistemov sestavljenih samo iz enega modela, ki ga je možno učiti od začetka-do-kraja. Sistem FR-zhch ima višjo uspešnost zaradi boljših rezultatov za `Explicit`-ne relacije (`Exp`). Pri tem celo presega prejšnji najboljši sistem od Schenk et al. [28], ki uporablja preprostejši model za `Explicit`-ne relacije. Imeti preprostejši model je lahko prednost na kitajščini, saj je na voljo bistveno manj `Explicit`-nih učnih primerov. Za ne-`Explicit`-ne relacije je zanimivo omeniti, da je na kitajski blind podatkovni zbirki uspešnost večine ostalih metod bistveno pod izhodiščnim modelom majority-zh. To namiguje, da sta sistema Schenk et al. [28] in Wang & Lan [21] preveč prilagojena na učno domeno in stil podatkovne zbirke CDTB. Po drugi strani pa oba naša sistema (FR-zh, FR-zhch) boljše ujameta ciljni koncept. Sistem FR-zh celo preseže naš prejšnji dvo-modelni sistem [22] za 3.6%.

Na angleški podatkovni zbirki v tabeli 5.7 oba naša sistema (FR-en, FR-ench) ne zaostajata veliko za najuspešnejšimi sistemi na blind podatkovni zbirki s 4.6%, čeprav ne uporabljata nobenega lingvističnega znanja ali zunanjih virov. Po pričakovanjih `Explicit`-ne relacije boljše klasificira Mihaylov & Frank [19], ki uporablja skrbno določen seznam diskurznih veznikov in značilk, povezanih z njimi. Sistem FR-en deluje mnogo bolje od sistema FR-ench za `Explicit`-ne relacije in posledično doseže višje ocene. Zato ker deluje na nivoju besed, se lažje nauči bolj smiselne besedne vložitve za predstavitev veznikov. Iz rezultatov ne-`Explicit`-nih relacij pa vidimo, da značilke od Mihaylov & Frank [19], ki temeljijo na pred-naučenih besednih vložitvah in med-argumentnimi podobnostmi, dosežejo prekomerno prilagajanje na učno domeno in stil podatkovne zbirke PDTB. Ta metoda ne deluje bistveno boljše od naše na blind podatkovni zbirki. Po drugi strani pa specializiran sistem od Rutherford & Xue [20] doseže mnogo boljše rezultate. Le-ta uporablja Bag-of-Words pristop z uporabo pred-naučenih besednih

vložitev in rezultati namigujejo, da lahko besedne vložitve boljše ujamejo ciljne koncepte šestih najpogostejših oznak smisla v angleščini. Vsekakor pa bistveno nižje F_1-ocene vseh sistemov na angleščini v primerjavi s kitajščino nakazujejo, da je klasifikacija smisla v angleščini bistveno težja kot v kitajščini. Razlike v velikosti podatkovne zbirke, slovnici, oznakah smisla in še posebej njihovi porazdelitvi močno vplivajo na uspešnosti.

Jezikovno-neodvisnost FR system smo potrdili z uspešno uporabo pretežno enakih hiper-parametrov modela na dveh bistveno različnih jezikih. Na kitajščini, kot primeru manj podprtega jezika, in na angleščini, kot najbolj raziskanem jeziku z najnaprednejšimi jezikovnimi tehnologijami.

Več podrobnosti lahko najdete v poglavju 5 in v članku Weiss & Bajec [23].

Zaključek

V disertaciji smo pristopili k najzahtevnejšemu delu besedilnega razčlenjevanja diskurza iz perspektive kako se otrok uči na podlagi primerov, brez eksplicitnega učenja. Namesto odvisnosti od ročno-izdelanih značilk, zunanjih virov in načrtovanjem sistema specifično za dani jezik in nalogo, mi stremimo k jezikovno-neodvisnemu pristopu za klasifikacijo smisla v plitkih diskurznih relacijah. V poglavju 3 najprej predstavimo našo focused RNNs plast, ki predstavlja nov tip plasti s pozornostnim-mehanizmom za izdelavo vložitev stavkov/argumentov. V poglavju 4 uporabimo te vložitve argumentov v našem FR system, ki predstavlja novo metodo za klasifikacijo smisla v plitkih diskurznih relacijah, ki temelji na focused RNNs plasti.

Najpomembnejša in unikatna značilnost FR system je, da se lahko skorajda brez sprememb uporabi na kateremkoli jeziku, oznakah smisla ter nivoju predstavitve vhodnih podatkov. To je možno, ker je sestavljen iz samo enega modela, ki ga je možno celostno učiti od začetka-do-kraja, namesto iz kompleksnega cevovoda bistveno različnih modelov. Njegovo jezikovno-neodvisnost smo potrdili z uspešno uporabo pretežno enakih hiper-parametrov modela na dveh bistveno različnih jezikih, a tudi s predstavitvijo vhoda tako na ravni besed kot na ravni znakov. Res je, da je potrebno model učiti na označenih primerih za vsak jezik, vendar ne potrebuje nobenih za jezik specifičnih značilk, cevovodov ali zunanjih virov, niti pred-naučenih besednih vložitev.

FR system smo primerjali z zmagovalnimi sistemi in močnimi izhodiščnimi modeli na kitajskem in angleškem jeziku z uporabo uradnih podatkovnih zbirk in metodologije iz CoNLL 2016 Shared Task [16]. Za 8% (z F_1-oceno 0.7477) presega ostale najboljše sisteme na kitajski blind podatkovni zbirki in ne zaostaja veliko (z F_1-oceno 0.5170) za

najuspešnejšimi sistemi na angleški blind podatkovni zbirki. Glede na to, da je angleščina najbolj raziskan jezik z najnaprednejšimi jezikovnimi tehnologijami, smo pričakovali, da bodo sistemi skrbno izdelani zanj bistveno prekašali naš jezikovno-neodvisen pristop.

Zaključimo z mislijo, da verjamemo, da je avtomatsko razčlenjevanje in analiza diskurza, še posebej klasifikacija smisla `Implicit`-nih diskurznih relacij, ključen naslednji korak pri razumevanju naravnega jezika. Čeprav teoretične osnove tega lingvističnega pojava še niso popolnoma razjasnjene, se je naš en model nevronske mreže sposoben naučiti potrebnih konceptov za klasifikacijo smisla brez ročno-izdelanih značilk in zunanjih virov. Poleg tega je verjetno, da bi večje količine učnih podatkov in naprednejši mehanizem bogatenja podatkov dvignili uspešnost FR system bližje človeškemu nivoju. Menimo, da takšen pristop ni samo koristen za avtomatsko klasifikacijo smisla v plitkih diskurznih relacijah, temveč bo navdihnil raziskovalce, da ga prilagodijo za kompleksnejše naloge iz NLP.