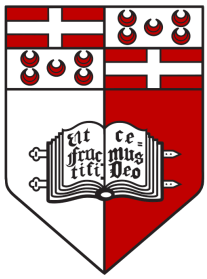
Cross-lingual Model Transfer for Neural Semantic Role Labeling

Jovana Urošević

MSc. Dissertation



Department of Intelligent Computer Systems
Faculty of Information and Communication Technology
UNIVERISTY OF MALTA

January 2019

Supervisors:
Dr Lonneke van der Plas, University of Malta
Dr Eneko Agirre, University of the Basque Country
Dr Barbara Plank, IT University of Copenhagen

Submitted in partial fulfillment of the requirements for the
Degree of European Master of Science in Human Language Science and Technology

M.Sc. (HLST)

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITY OF MALTA

Declaration

Plagiarism is defined as "the unacknowledged use, as ones own work, of work of another person, whether or not such work has been published" (Regulations Governing Conduct at Examinations, 1997, Regulation 1 (viii), University of Malta).

I, the undersigned, declare that the Master's dissertation submitted is my own work, except where acknowledged and referenced.

I understand that the penalties for making a false declaration may include, but are not limited to, loss of marks; cancellation of examination results; enforced suspension of studies; or expulsion from the degree program.

Student Name: Jovana Urošević
Course Code: CSA5310 HLST Dissertation
Title of work: Cross-lingual Model Transfer for Neural Semantic Role Labeling

Signature of the student:

Date: January 4th, 2019

Acknowledgements

First and foremost, I would like to express my profound gratitude to my supervisor at the university of Malta, Lonneke van der Plas, for her endless patience and invaluable advice throughout this whole process.

Then, I would like to thank Barbara Plank for all the great explanations of complex neural matters which were so new to me, and for the encouragement to try to do a project which I thought was above my abilities.

Also, a big thank you to Eneko Agirre for his help throughout this project, as well as great lectures on computational semantics which made a big influence on my current research interests.

I would like to thank my mum and my sisters for their support during these two years of the course. You all sacrificed a lot for me and thank you for that.

I would also like to thank Kenny for being a great companion during the time of struggle, and for all the great time we had together in Malta and Spain.

And finally, I would like to thank André for so much love and support during this period. You know that it would be so much more difficult to finish this if it was not for you. There, you have it written now :)

Abstract

Semantic role labeling (SRL) is an essential task for understanding natural language, which allows for identifying events and their participants in a text by means of semantic roles (eg. agent, theme, goal). As such, it has been proven useful for a wide range of tasks in natural language processing, such as information extraction, question answering, machine translation, etc. However, the amount of manually labelled data necessary for building SRL systems is unfortunately not available for most languages given the time and resources required for their creation. Various cross-lingual methods have been suggested in order to create such models for resource-poor languages by means of model transfer or annotation projection while often making use of existent monolingual SRL systems and parallel language corpora.

The proposed thesis builds upon the work of Kozhevnikov and Titov (2013) who proposed a model transfer method for SRL of English, French, Czech and Chinese languages by making use of shared feature representations, such as cross-lingual clustering or cross-lingual distributed word representations, and machine learning. Even though they have demonstrated competitive results without using word-aligned parallel corpora, they still make use of syntactic information for the transfer. In this project, we show that by using a state-of-the-art neural SRL system (He, Lee, Lewis, et al., 2017) and pretrained cross-lingual word embeddings (Smith et al., 2017) we can achieve competitive results even without using any syntactic information. On the CoNLL-2009 data, our best models achieve weighted F1 score of 74.44 and 79.86 for French and Chinese language, respectively.

Key words: semantic role labelling, model transfer, cross-lingual word embeddings, deep learning, bi-LSTM

Contents

List of Figures

List of Tables

List of Abbreviations

Bi-LSTM	Bi-directional Long Short-term Memory
CBOW	Continuous Bag-of-words
CoNLL	Conference on Natural Language Learning
CRF	Conditional Random Field
DEEPATT	Deep Attentional Neural Network
DSMs	Distributional Semantic Models
FEs	Frame Elements
FFN(s)	Feed-forward Network(s)
GloVe	Global Vectors for Word Representation
LSTM	Long Short-term Memory
ML	Machine Learning
MLP	Multi-layer Perceptron
MSE	Mean Squared Error
NLP	Natural Language Processing
NLU	Natural Language Understanding
NN(s)	Neural Network(s)
OOV	Out-of-vocabulary
PMI	Positive Pointwise Mutual Information
POS	Part-of-speech
PropBank	Proposition Bank
RNN	Recurrent Neural Network
SRL	Semantic Role Labeling

Chapter 1

Introduction

Semantic role labeling (SRL) is the task of automatically assigning labels to predicates and their arguments in a sentence which are indicators of their shallow semantic roles, such as those of agent, theme, goal or other. A predicate is usually evoked by a certain word or phrase, and is accompanied by one or more arguments which have different (semantic) roles in sentences depending on the evoked predicate. This kind of analysis is essential for the task of natural language understanding (NLU), since it allows meaning to be extracted out of possibly syntactically different sentence structures. For example, taking a look at the annotated sentences in example 1 below, we can see that the meaning (and thus semantic roles) stay unchanged across different syntactic structures. The predicate is marked by letters in bold, while each of the arguments is in square brackets with its semantic role explicitly marked.

(1) a. [$_{Agent}$ Jessica] [$_{Rel-load.01}$ **loaded**] [$_{Theme}$ boxes] [$_{Destination}$ into the wagon].

b. [$_{Agent}$ Jessica] [$_{Rel-load.01}$ **loaded**] [$_{Destination}$ the wagon] [$_{Theme}$ with boxes]. [1]

This kind of analysis enables a range of NLP tasks, some of which are the following: information extraction (Fader et al., 2011), question answering (Kaisser and Webber, 2007, Shen and Lapata, 2007, Maqsud et al., 2014), machine translation (Liu and Gildea, 2010, Gao and Vogel, 2011, Lo et al., 2013), dialogue systems (Van der Plas, Henderson, et al., 2009), text summarization (Trandabât, 2011, Khan et al., 2015), opinion mining (Marasović and Frank, 2018). Therefore,

[1] Example taken from Van der Plas, Apidianaki, et al. (2014).

several invaluable semantic resources have been created (such as FrameNe[2], PropBank[3], NomBank[4], VerbNet[5], etc.) which allow us to carry out many of these natural language processing tasks that involve meaning. All of these resources, however, have been developed by linguists' manual efforts over an extensive period of time and are most commonly available solely for the English language. Also, the amount of manually labelled data necessary for building SRL systems is unfortunately not available for most languages given the time and resources required for their creation. That is why various cross-lingual methods have been suggested in order to create such models for resource-poor languages.

In recent years, there have been attempts to create unsupervised (Lang and Lapata, 2010, Titov and Klementiev, 2012b, Luan et al., 2016, Titov and Khoddam, 2015), semi-supervised (Deschacht and Moens, 2009, Fürstenau and Lapata, 2012, Zadeh et al., 2011), transfer systems (Kozhevnikov and Titov, 2013), as well as annotation projection systems for SRL (Padó and Lapata, 2009, Van der Plas, Samardzic, et al., 2010, Van der Plas, Apidianaki, et al., 2014, Akbik, Chiticariu, et al., 2015). All of these methods share an effort to benefit from unlabeled, monolingual data or some type of cross-lingual resources, such as parallel corpora, instead of relying on large amounts of often unavailable annotated data.

Looking more closely into this research direction, unsupervised methods seem the most appealing for the low-resource setting, since they require the least amount of multi- or cross-lingual data, and are supposed to be language-independent. However, their performance is still not able to surpass those of the transfer and annotation projection models (Kozhevnikov and Titov, 2013). On the other hand, there has been an extensive amount of research conducted to investigate the possibility of automatically projecting semantic role labels from resource-rich languages (in most cases English) to resource-poor languages by using word-aligned parallel corpora. The underlying assumption behind this idea is that translated sentences in parallel corpora are semantically equivalent to the original sentence, given that they represent their translations. In this way, resources such as the ones mentioned earlier can automatically be created and enable the training of statistical SRL systems for new languages. This relatively simple method, however, suffers from incorrect word alignments, structural differences between languages, and in addition performs badly when applied to a domain it was not trained on (Akbik, Chiticariu, et al., 2015). And finally, research done on SRL model transfer has

[2]FrameNet (Fillmore and Baker, 2009): https://framenet.icsi.berkeley.edu/fndrupal/fnbibliography

[3]PropBank (Palmer et al., 2005): https://propbank.github.io/

[4]NomBank (Meyers et al., 2004): https://nlp.cs.nyu.edu/meyers/NomBank.html

[5]VerbNet (Schuler, 2005): https://verbs.colorado.edu/verbnet/

achieved better results when compared to both annotation projection and unsupervised methods by taking a system trained on one language and applying it to another (Kozhevnikov and Titov, 2013). These systems usually either use delexicalized models leveraging only formal shared features across languages (such as POS-tags for example), or use shared lexical feature representations for the languages involved, such as cross-lingual clustering or cross-lingual distributed word representations (Kozhevnikov and Titov, 2013). This approach showed competitive results, even when compared to the supervised state-of-the art systems that make heavy use of annotated data, and as such show that similar cross-lingual approaches could have an important role for natural language processing in low-resource settings.

The proposed thesis is aimed to improve the results obtained by Kozhevnikov and Titov (2013) by transferring the deep SRL model of He, Lee, Lewis, et al. (2017) and making use of pretrained cross-lingual word embeddings (Smith et al., 2017). A similar approach has been successfully applied before for the model transfer of the universal dependency parser by J. Guo et al. (2016), who showed that using cross-lingual word embeddings reduces the lexical gap between different languages by projecting their monolingual embeddings to a common vector space. On top of that, they show that one of the key components for the success of their results is the use of a deep bi-directional Long Short-term Memory Network. Given that this approach has given improvements over other traditional methods when transferring a dependency parser, we are expecting similar outcome for our deep SRL model transfer.

1.1 Motivation and Research Objectives

The research objectives of this thesis are split in five categories we wanted to explore and will be explained in more detail further below.

- *Research question 1:*
  - Can a system based on a deep neural network outperform the results obtained with the traditional machine learning approach used by Kozhevnikov and Titov (2013) for the task of SRL model transfer for the English, Chinese and French language?
  - Given that their model does not always outperform the annotation projection method used as a baseline, the second question is consequently: can our system also outperform the annotation projection model of

Kozhevnikov and Titov (2013)?

Our *main* objective is to use the deep SRL system of He, Lee, Lewis, et al. (2017) and incorporate the pretrained corss-lingual word embeddings of Smith et al. (2017) for different languages in place of the initial monolingual English embeddings, and in such way train and make available SRL systems for the Chinese and French language. The goal is to try and outperform the currently available transfer system of Kozhevnikov and Titov (2013) for the mentioned languages, as well as the results they reported for the annotation projection system. Besides currently being the most dominant approach, neural networks are also producing state-of-the-art results in a range of NLP tasks, such as language modelling (Yang et al., 2017; Krause et al., 2018), POS-tagging (Ling et al., 2015; Bohnet et al., 2018), dependency parsing (Dozat and Manning, 2016; Clark et al., 2018), etc. On top of that, the recent approach of J. Guo et al. (2016) to the task of model transfer of a dependency parser has shown that the combination of cross-lingual embeddings with a deep neural network approach significantly outperforms previously suggested traditional machine learning approaches. Given that their method showed promising results, we would expect a similar outcome for the model transfer of semantic role labels.

- *Research question 2 - Syntactic information:*
  - How does the model perform without any syntactic information as compared to the previous work?

We would like to train and test the system without any use of syntactic information just like the original system of He, Lee, Lewis, et al. (2017) does. There has been a range of SRL models suggested in the past that make use of different types of syntactic information, which are often part-of-speech tags, syntactic treebanks or a combination of both. However, even though it has been shown previously that syntactic information can benefit the task of shallow semantic parsing (Màrquez et al., 2008; Roth and Lapata, 2016), this type of information is not always available for different language and in the task of model transfer it was shown that transferred syntactic information could be one of the sources of errors for structurally more distant languages (Kozhevnikov and Titov, 2013). Finally, in this thesis, we wanted to follow the recent trend in SRL that tries to rely on as little syntactic data as possible (such as in He, Lee, Lewis, et al. (2017), Tan et al. (2018), and He, Lee, Levy, et al. (2018)).

- *Research question 3 - Language similarity:*
  - Does incorporating only structurally similar languages to a common

vector space result in a boost in performance?

Since there are pretrained multilingual embeddings available for Romance languages, we would like to see how the result of our system for French changes when it uses more similar languages projected in the common vector space. Does incorporating more languages result in a boost in the performance or does it just introduce more noise to the system?

- *Research question 4 - Word embeddings alignments:*
  - Does using monolingual word embeddings for the three languages aligned to a common vector space with different techniques make a significant difference in the obtained results?

According to the recent survey of cross-lingual embeddings by Ruder et al. (2017), different pretrained embeddings were evaluated for specific tasks and do not guarantee the same results when used in other tasks or with different systems and data. Therefore, we would like to compare two types of readily available cross-lingual embeddings aligned using the approaches suggested by Smith et al. (2017) on the one hand and by Conneau et al. (2018) on the other. These approaches will be both described in detail in the Chapter 4.

All or combination of the above mentioned ideas and research objectives are supposed to produce a successful SRL system for the two target languages (French and Chinese), but is ideally possible to modify and apply to any other language as long as monolingual embeddings and seed dictionaries are available for it. And this is exactly the motivation for building such a system - make available a method for obtaining shallow semantic parsers for languages other than English at fairly a low cost. In the future, if this method shows good results, we would like to apply it in a real low-resource setting as well, in which no available syntactic information nor labelled data is available.

1.2 Outline

The structure of the thesis is as follows. The Chapter 2 covers the theoretical background of the thesis. Firstly, it introduces the framework of semantic roles and the task of semantic role labeling (2.1), followed by an overview of different approaches to SRL (2.1.1), which cover supervised methods as well as the ones applied in a low-resource scenario. Finally, vector semantics is described in the Section 2.2 followed by a brief introduction to common neural network architectures used in NLP research (2.3) with more detail about the Recurrent Neural

Networks that we will make use of in this thesis.

Furthermore, Chapter 3 covers the related work, presenting the relevant research done on syntax-agnostic neural SRL approaches (3.1), SRL model transfer (3.2) as well as related methods in the area of nerual model transfer for syntactic dependencies (3.3).

Following, in Chapter 4, we present the design and the methodology that was followed in order to create our neural SRL transfer model. We start by describing the system of He, Lee, Lewis, et al. (2017) used for the transfer, the cross-lingual embeddings alignment approaches (4.2), the data that we used to train and evaluate our models on (4.2), and finally we give an overview of the implementation and evaluation methodology (4.4 and 4.5, respectively).

In Chapter 5, we describe all the different experiments that have been conducted, and the obtained results and their analysis. The analysis is then further expanded in the Section 5.5, which gives a brief discussion of the results and the systems' performance, as well as of the spotted errors our parser tends to make for different languages.

And finally, Chapter 6.2, summarizes the most important findings of the experiments we carried out while reflecting both on advantages and disadvantages of our models. Based on the results we have obtained, we also give suggestions for future work that could further explore similar methods for low-resource SRL and potentially lead to an improvement in the accuracy of the models suggested in this thesis.

Chapter 2

Theoretical Background

In this chapter, we will give a brief overview of the theoretical background needed to understand the systems we use in this project. In order to do this, the chapter is split in two main parts. On the one hand, we introduce the theory behind semantic role labeling, including the common approaches, resources and annotation used to build such systems. And on the other hand, we introduce vector semantics and neural network approaches that have recently become essential parts of natural language processing and which we will make use of in this project as well.

2.1 Semantic Role Labelling (SRL)

In order to claim that we have understood a particular event, we need to be able to answer the following questions concerning it: "Who did what to whom (when, where and how)?" (Palmer et al., 2005). The words "who, what, to whom" are essential because they represent the core participants of an event, while the other three are optional as they provide additional information. If we take a look at example 2 below, we will see that we are able to answer the previous questions (or at least most of them) with the same answers even though the same event might have been expressed by syntactically different structures.

(2) a. **Chuck bought the car.**

[$_{Who}$ Chuck] [$_{What}$ bought] [$_{To\_whom}$ the car].

b. **The car was bought by Chuck**.

[$_{To\_whom}$ The car] [$_{What}$ was bought] [$_{Who}$ by Chuck].

c. **The purchase of the car by Chuck...**

[$_{What}$ The purchase] [$_{To\_whom}$ of the car] [$_{Who}$ by Chuck]...

d. **The car was sold to Chuck by Jerry**.

[$_{To\_whom}$ The car] [$_{What}$ was sold] [$_{To\_whom}$ to Chuck] [$_{Who}$ by Jerry].

e. **Jerry sold the car to Chuck.**

[$_{Who}$ Jerry] [$_{What}$ sold] [$_{To\_whom}$ the car] [$_{To\_whom}$ to Chuck].

We see that the sentences can have a different word order, can be in a different voices (passive, active), that the action can be expressed by different verbs (*buy, sell*) or nouns (*purchase*), etc. and yet, they all still convey essentially the same meaning and answer the same questions.

The idea behind this kind of meaning decomposition is that we have predicates (mostly verbs, but also nouns, some adjectives, adverbs, phrases) that require certain arguments which can carry a particular (semantic) role with respect to the predicate. Semantic roles should allow us to extract meaning across differently structured sentences, because the meaning stays the same no matter the surface form we express it with. Semantic roles show us that there are several participants in the previously mentioned purchase event in example 2: *Chuck, Jerry, the car* and that each of them has a specific role in the sentences. This kind of meaning representation is called *shallow semantic parsing* and it uses semantic roles to mark pieces of meaning in a sentence (Jurafsky and Martin, 2018). The roles and their names are often closely related to their syntactic function in the sentence. For example, *Chuck* is a subject in the first sentence and has a role of an agent; *the car* is an object and has a role of a patient, etc. This could be taken as a general rule: agents are often subjects of a sentence, direct objects are the theme, etc. (Jurafsky and Martin, 2018). There are, however, exceptions to these rules, which are described further below and illustrated in example 3.

The idea of thematic roles, or marking participants and events in a sentence, is a very old one in linguistics, but there is still no fixed set of thematic roles researchers agree on. Thus, we will see different sets of roles from one paper to another. Depending on the author, semantic roles can range from more specific (as BUYER role), more abstract ones (AGENT role), to those on a very abstract level (as PROTO-AGENT) (Jurafsky and Martin, 2018). Therefore, some authors will have very specific roles and hence quite many of them, while others will try to use fewer but more general roles. In Table 2.1, obtained and adapted from Jurafsky and Martin (2018), some of the most commonly used thematic roles

Thematic Role	Definition	Example
Agent	The volitional causer of an event	*That girl* plays tennis.
Experiencer	The experiencer of an event	*Mary* felt ill.
Force	The non-volitional causer of the event	*The wind* blows debris into.
Theme	The participant most directly affected by an event	The boy kicked *the ball*.
Result	The end product of an event	The city built *a new park*.
Content	The proposition or content of a propositional event	Mat said *"I can help you!"*
Instrument	An instrument used in an event	He signed *with a black pen*.
Beneficiary	The beneficiary of an event	They made *me* a surprise.
Source	The origin of the object of a transfer event	I flew in *from Madrid*.
Goal	The destination of an object of a transfer event	I drove *to Rome*.

Table 2.1: Commonly used thematic roles, their definitions and examples.

are represented. The term semantic roles is normally used to generalize over all different sets of thematic roles that different researchers use.

In computational linguistics, semantic roles are used for the task of *semantic role labeling* (SRL), which refers to the task of automatically assigning semantic roles to constituents or phrases in a sentence, that is, to find the predicates and assign roles to their arguments (Jurafsky and Martin, 2018). Having these labels allows us to infer sentence meaning that we might not always be able to infer from a syntactically parsed sentence. For example, while the AGENT is often realized as the subject of the sentence, in other cases the THEME can be the subject (Jurafsky and Martin, 2018). The following are possible realizations of the thematic roles of the verb *break*:

(3) a. [$_{Agent}$ John] **broke** [$_{Theme}$ the window].
b. [$_{Agent}$ John] **broke** [$_{Theme}$ the window] [$_{Instrument}$ with a rock].
c. [$_{Theme}$ The window] **broke**.
d. [$_{Theme}$ The window] **was broken** [$_{Agent}$ by John]. [1]

Each predicate has a set of arguments that can accompany it called the *thematic grid* or *case frame* (Jurafsky and Martin, 2017). The case frame basically presents different ways of describing the same situation or an event by a certain predicate. In the above example, the case frame of the verb *break* is composed of AGENT, THEME and INSTRUMENT, but there are other roles that can appear, too. Besides, some predicates allow their arguments to have a more free word order:

(4) a. [$_{Agent}$ Doris] **gave** [$_{Theme}$ the book] to [$_{Goal}$ Cary].
b. [$_{Agent}$ Doris] **gave** [$_{Goal}$ Cary] [$_{Theme}$ the book]. [2]

[1] Example taken from Jurafsky and Martin (2018).
[2] Example taken from Jurafsky and Martin (2018).

These are called *verb alternations* or *diathesis alternations* (Jurafsky and Martin, 2018). On top of this, it is possible that a predicate requires different set of semantic roles depending on its sense in a particular context. For example, if we take a look at the two senses of *buy* (Palmer et al., 2005) in the example 5, we will see that the sets of roles defined for them are different.

(5)

buy.01: "purchase"	***buy.05: "accept as truth"***
Arg0-PAG: *buyer*	Arg0-PAG: *believer*
Arg1-PPT: *thing bought*	Arg1-PPT: *thing believed*
Arg2-DIR: *seller*	
Arg3-VSP: *price paid*	
Arg4-GOL: *benefactive*	

All of the previously mentioned facts, make it clear that automatically disambiguating predicates and accordingly recognizing and labeling their arguments is a very challenging task. This resulted in, on the one hand, vast research done on trying to develop corpora labelled with semantic roles which would then enable, on the other hand, training of automatic SRL systems.

When it comes to the development of different resources, it has been a problem to define a fixed set of roles with their definitions and the inventory of case frames that all researchers would use. This lead to introducing different semantic role models that either have more specific or more general roles. Among most known resources for shallow semantic parsing are FrameNet (Fillmore and Baker, 2009), VerbNet (Schuler, 2005), PropBank (Palmer et al., 2005) and its extension NomBank (Meyers et al., 2004). We will not describe these resources here, but for a very good survey of most of the state-of-the-art semantic resources we refer to Abend and Rappoport (2017). For the purpose of our thesis, we will make use of the PropBank annotation, which is currently most commonly used resource for training supervised SRL systems.

Given the difficulties SRL task imposes, besides developing different resources, many different approaches to the task have been suggested over the years as well. We will give a brief description of them in the following sections by dividing them into the ones that always require labelled resources (supervised approaches) and the ones that are trying to use less or as little labelled data as possible (approaches applied in lower-resources scenarios).

2.1.1 Different Approaches to SRL

This section will cover different approaches to semantic role labeling, with a particular focus on SRL methods in low-resource settings. Here, we will shortly cover a broader theoretical description of the approaches, while in section 3 we will give a more detailed overview of those most relevant to our topic.

2.1.1.1 Supervised Learning Methods

The emergence of resources such as FrameNet and PropBank enabled a new wave of research for SRL. Previous rule-based methods were abandoned in favor of supervised machine learning given the new, large corpora with reliable and hand-annotated semantic roles (Jurafsky and Martin, 2018). One of the first and well-known statistical SRL algorithms was trained by Gildea and Jurafsky (2002) using the English version of FrameNet. In order to identify the constituents and assign them with correct roles, they applied statistical methods, such as probabilistic parsing and statistical classification. The final result was a method that assigns semantic roles to sentence constituents with a 77% accuracy that could then be used in various NLP applications. Furthermore, one of the first semantic parsers that used PropBank came soon after and was created by Gildea and Palmer (2002). On top of that, this paper pointed out the importance of accurate syntactic parses for the task of SRL.

These first SRL systems set the base for a lot of the systems that came after, especially when it comes to the used feature set. Most of the traditional supervised learning approaches use the features proposed by Gildea and Jurafsky (2002) or some kind of generalization of their suggested features (Jurafsky and Martin, 2018, Màrquez et al., 2008). These typically include: the governing predicate, phrase type of the constituent (noun phrase, prepositional phrase, etc), headword of the constituent, POS of the headword, the path from the constituent to the predicate in the parse tree, voice of the sentence or a clause in which the constituent appears (active or passive), subcategorization pattern of the predicate (the set of arguments that is likely to appear in the phrase type of the predicate), the position the constituent has in the sentence relative to the predicate (left or right). Each of these features has, of course, later been expanded and further developed by other researchers , such as Surdeanu et al. (2003), Nianwen Xue and Palmer (2004), Pradhan, Ward, et al. (2005), Zapirain et al. (2007), and so on.

Besides the typical set of features, a typical SRL pipeline that is used in most of the supervised systems in one way or another can be defined. Given a sentence and

its syntactic parse tree, algorithms need to, given a predicate, identify boundaries of its arguments (*argument identification*) and label them with a correct role (*argument classification*) (Màrquez et al., 2008). This is normally done in three steps: (1) filtering (pruning) a set of possible arguments for the predicate; (2) local scoring, where each argument candidate is assigned a confidence score for each possible role label; and (3) global scoring, where local scores are combined in order to produce the most probable sequence of labelled arguments given the predicate (Màrquez et al., 2008).

Recent work on supervised SRL is mostly focused on applying deep neural networks (NNs). One of the biggest differences between these models and the traditional ones is that the traditional approaches needed to rely on complex sets of hand-crafted input features, which in most cases heavily relied on syntax. However, NN approaches have slowly moved away from this tradition. One of the first SRL NN models was by Collobert et al. (2011), but it did not manage to outperform the traditional models. One of the first successful application of neural networks in a SRL task was by FitzGerald et al. (2015) who used arguments and semantic roles jointly embedded in a shared vector space as input features for a given predicate. Even though syntactic information was considered important for SRL and used in SRL tasks for a long time, most recent work has proven the opposite by achieving state-of-the-art results without using any syntactic information (for example, Zhou and Xu, 2015, He, Lee, Lewis, et al., 2017, Marcheggiani, Frolov, et al., 2017). Therefore, we could say the NN methods could be roughly split into syntax-aware (for example, FitzGerald et al., 2015, Roth and Lapata, 2016, Marcheggiani and Titov, 2017) and syntax-agnostic methods (for example Zhou and Xu, 2015, He, Lee, Lewis, et al., 2017, Marcheggiani, Frolov, et al., 2017) based on the importance they place on syntax for SRL and how much syntactic information they use, if any. Given that we have a multilingual, low-resource setting in mind for the current project, the systems that make use of a lot of syntactic information (such as fully parsed sentences) are not very relevant, because that kind of information is difficult to obtain for many languages. The systems that are not using syntax at all or are limited to using only POS-tags are covered in the related work section (Chapter 3) since their methods are more suitable for our project. However, a good overview of the most recent, deep neural approaches is given in Marcheggiani, Roth, et al. (2017).

2.1.1.2 SRL in a Low-resource Context

Most of the research in SRL falls under the umbrella of supervised machine learning approaches and, thus, relies on hand-annotated data for semantic roles in order to train such systems. This is especially true with the recent emergence of deep neural approaches, which require even larger amounts of data for training. Such methods achieve fairly good performance, mostly around 80% of the F1 measure on the standard test collections for the English language (Kozhevnikov, 2017). However, these are predominantly developed for English, as obtaining labeled data for supervised learning is expensive and time-consuming, and, therefore, prevents the development of (better) SRL systems across different languages. Data for other languages is not non-existent, but it tends to be smaller than that of English and sometimes it is of lower quality. The lack of (high-quality) annotated data has given rise to several approaches that try to avoid or reduce the need for such data.

2.1.1.2.1 Unsupervised Methods. We have seen that even a small amount of labelled data could be difficult to obtain. Thankfully, other methods and approaches have been explored in order to completely avoid the need for labelled data. One such direction is the research conducted in the area of unsupervised learning, which makes assumptions about the semantic roles of arguments of a certain predicate based on the generalizations about arguments' syntactic functions (Jurafsky and Martin, 2018). This implies the assumption that the system can learn a lot from just looking at the syntactic features of arguments. For example, subjects often tend to have a semantic role of an agent, while objects lean to fill in the roles of a theme or an instrument, etc. It is important to notice that this is not always the case, but rather just a good place to start, a good baseline (Kozhevnikov, 2017). Besides taking syntax into account as a feature, there have been attempts of including information about the characteristics of predicates (voice, for example), information about the word order in a sentence, some lexical data, etc. (Kozhevnikov, 2017).

One of the first unsupervised systems for SRL were firstly proposed by Swier and Stevenson (2004) and then followed shortly by that of Grenager and Manning (2006). These models relied on statistical methods and syntactic features mentioned above, which they applied to VerbNet and PropBank, respectively. Building upon this work, Lang and Lapata (2011a, 2011b) propose two methods - induction via split-merge and with graph partitioning methods. Shortly after, Titov and Klementiev (2012a) introduce Bayesian methods to solve the unsuper-

vised SRL task, which lead to state-of-the-art performance. They also tried using parallel data to improve the results (Titov and Klementiev, 2012b), as well as a small amount of labeled data, and in this way strive to incorporate semi-supervised methods to help augment the performance of their unsupervised method (Titov and Klementiev, 2012c).

As is the case with most of the recent work, newer research in unsupervised learning also involves using neural architectures and embeddings. Titov and Khoddam (2015) suggest a system based on autoencoders, which performed equally well as the previous, traditional unsupervised methods only without using any kind of linguistic information about the languages involved. Finally, Luan et al. (2016) tried to treat the problem of unsupervised SRL as an argument embedding problem achieving state-of-the-art performance.

In summary, unsupervised methods seem very appealing for the low-resource setting since they require the least amount of multi- or cross-lingual data among all of the approaches that will be mentioned. They are also language-independent as some of the research on dependency parsing has shown (for example, Jiang et al., 2016, Cai et al., 2017 etc.). However, their performance is still not able to beat those of the transfer and annotation projection models (Kozhevnikov and Titov, 2013) which we will describe next.

2.1.1.2.2 Cross-lingual Annotation Projection. In the previously described approaches none or very little data was used. However, this and the following section will cover those approaches that make use of existing labelled corpora for developing SRL systems languages other than English. Such approaches are *cross-lingual annotation projection* and *model transfer*.

Cross-lingual annotation projection tries to transfer semantic roles from a resource-rich source language (such as English) to a resource-poor target language, where the source language side is already annotated either manually or automatically. Most often, researchers use either FrameNet or PropBank and parallel corpora. The idea here is to use some type of alignment between sentences in the two languages, either by aligning tokens (words), constituents or whole sentences. After having obtained the alignments, the annotations is then transferred from a source text unit to the target one. This method is also called *direct transfer*, because the annotations are being transferred directly from one unit to another (Van der Plas, Merlo, et al., 2011).

In order to do this, there are several conditions that need to be met: we need to have an annotated corpus for the source language to train an SRL model, a

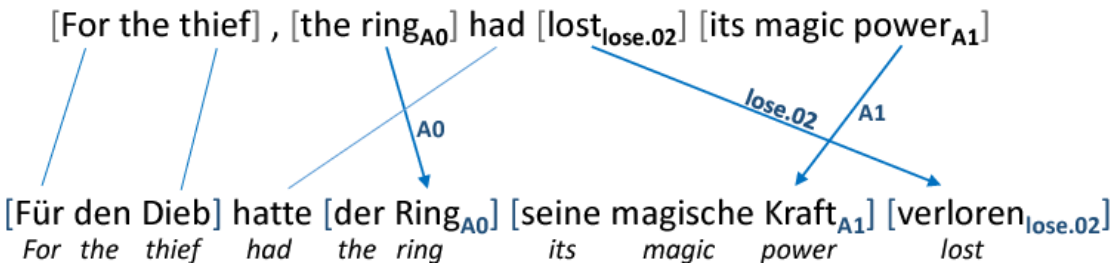


Figure 2.1: Example of an annotation projection from English to German using word alignments in parallel sentences (Akbik, Guan, et al., 2016).

parallel corpus with semantically equivalent sentences in two languages, and on top of that, in order to obtain correct transfers, we need to have correct and complete alignment pairs between the languages (Van der Plas, Merlo, et al., 2011, Van der Plas, Apidianaki, et al., 2014). When this is fulfilled, we can use an existing SRL model to annotate the source with labels and then, using alignments, transfer those labels to sentences in the target language (example in figure 2.1). In this way we can obtain an automatically labeled target language corpus with semantic roles, which enables the training of SRL systems for target languages (Akbik, Chiticariu, et al., 2015, Van der Plas, Merlo, et al., 2011).

At first, attempts of direct transfer of FrameNet frames were made, for example in Padó (2007), Padó and Lapata (2009), Basili et al. (2009), Annesi and Basili (2010). However, later work has shifted to PropBank because of its broader coverage and because it is more suitable to use in combined semantic-syntactic settings (Merlo and Van Der Plas, 2009, Van der Plas, Merlo, et al., 2011). However, even though the direct transfer of annotations is easy to implement for any language pair, the problem of missing or incorrect alignments between pairs caused by non-literal translations or *translation shifts* (also called *translation divergences*) are the main cause of wrongly transferred or missing annotations, as stated in many of the previously conducted research (such as Padó, 2007, Van der Plas, Merlo, et al., 2011, Van der Plas, Apidianaki, et al., 2014, Akbik, Chiticariu, et al., 2015, Aufrant et al., 2016). Therefore, the situation seen in examples like the aforementioned one, where we have completely correct, token-to-token alignments, is not always existent.

More recent work has, thus, focused on solving this problem of alignments. In Van der Plas, Apidianaki, et al. (2014), a different approach was suggested that does not use any parallel data, but instead gathers alignment information from the whole corpus. In this way, the system is not affected by non-literal translations and reduces the amount of errors caused by wrong alignments because it simply does not depend on the token-to-token correspondences (Van der Plas, Apidianaki,

et al., 2014). Here, the transfer is modeled as a word sense disambiguation task, where French verbs are annotated with English predicate labels (Van der Plas, Apidianaki, et al., 2014). Building upon this work, Akbik, Chiticariu, et al. (2015) also suggests a way of avoiding above listed common errors by creating a model that iteratively projects annotation only for those arguments and predicates with high-confidence. In order to show that this model is language-independent, the authors successfully applied it to seven different languages.

However, the most common problems to this day occur because of translation shifts which represent cross-lingual differences coming to light when a translation of one language into another results in a very different structure than that of the original. Clearly, these kinds of differences are expected and will always happen in any type of parallel corpora, no matter how controlled the language in them is.

2.1.1.2.3 Cross-lingual Model Transfer. The second approach that makes use of existing labelled corpora of resource-rich languages is, the so called, *model transfer*, and represents an approach of, in simple words, training a model on one language and, with certain modifications, applying it directly to another language. It was initially introduced by Zeman and Resnik (2008) for the purpose of syntactic parsing. This type of approach has a potential of being less affected by parallel data and wrong alignments, since it learns a shared representation between the languages involved (Kozhevnikov, 2017). This in fact means that we would represent the information we want the system to process in a way that it would be shared between the languages involved and that the system processing this information would not be able to distinguish between the languages in question. Like this, we would then train the system on one language and apply it directly to another. For example, Zeman and Resnik (2008) experimented on the Danish-Swedish language pair. Even though these languages are very similar both lexically and syntactically, they still tend to use different spellings for the same words, and thus, present an obstacle to the system which would then mark most of the words as unknown. As a type of shared representation for these two languages, they used glosses and POS-tags. With the former, the model is trained on Danish and then, when tested, the Swedish words are replaced with glosses - Danish translations with the highest weight. Like this, the system can recognize all the words and is able to parse the test set, after which the original Swedish words are again introduced and the success of the parser is finally evaluated. The latter shared representation, however, removes every lexical information from the training data and relies only on the POS-tags for training, and the same is done for testing. This approach of removing lexical information is called *delexicalization*.

Besides these two types of shared features, other approaches were proposed, also for syntactic parsing, such as mapping words from both languages to a cross-lingual word-cluster (Täckström et al., 2012) or using a distributed word representation (Klementiev et al., 2012, Xiao and Y. Guo, 2014, Vulić, 2017). It is important to notice that even though these methods are more resistant to the already mentioned difficulties the parallel data brings, they still need some type of knowledge about the connection between the shared information between languages involved. However, they do not require complete and accurate alignments of words, phrases or sentences as annotation projection approaches do.

All of these ideas have developed further in the area of dependency parsing, which can be seen in more recent work (section 3) that involves exploring the combination of deep neural approaches and different types of distributed word representations, such as in Xiao and Y. Guo (2014), Duong et al. (2015b), J. Guo et al. (2016).

For the task of SRL, Kozhevnikov and Titov (2013) have made an attempt of transferring semantic roles from resource-rich to resource-poor languages. Since our work builds directly on their ideas, this and other related research will be covered in detail in the following chapter (Chapter 3).

2.2 Vector Semantics and Word Embeddings

Even though the use of word embeddings became popular in the field of Natural Language Processing in the recent years, the idea of representing words in a language as vectors in a high dimensional space goes far back to a semantic theory called Distributional Hypothesis which was first introduced by linguists Harris (1954)) and Firth (1957) (Jurafsky and Martin, 2018). Distributional hypothesis is best known by Firth's quote: *you shall know a word by the company it keeps*. This refers to the fact that words similar in meaning tend to appear in similar contexts. For example, *book* and *article* will often appear near words such as *read, write, publish, text, story*, etc. To put it differently, words that have similar semantic characteristics, tend to have similar distributional properties.

This idea was then adopted by Distributional Semantics in which computational models are built to automatically learn word representations by looking into their distribution in a large amount of text (corpora). In this way, we represent words' meaning as a set of contexts they appear in (Jurafsky and Martin, 2018). In order to find words with similar meaning, we just need to compare their distributions, i.e. co-occurrence counts. After obtaining co-occurrences, we are able to represent word meaning as a vector or a point in a high dimensional space. Closely related

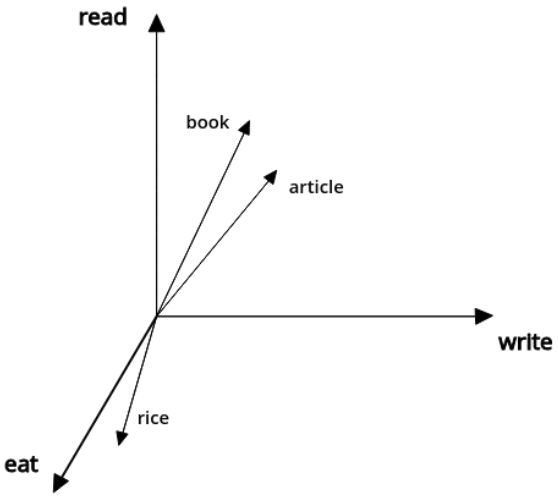


Figure 2.2: Vector space model with three dimensions.

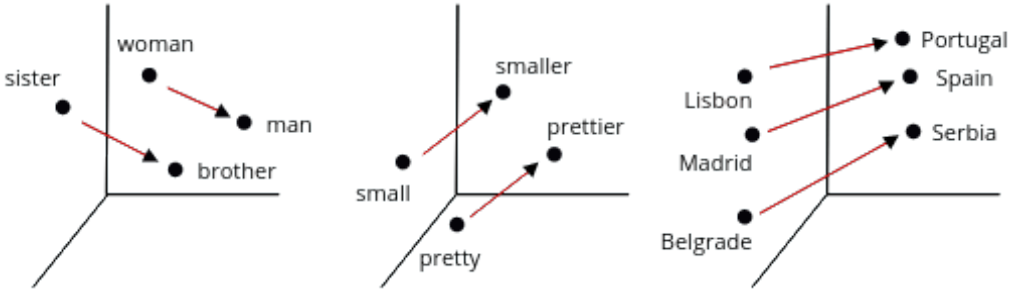


Figure 2.3: Semantic and syntactic relations captured as linear relations in a simplified vector space models.

words will tend to be closer together in the semantic space, as we have shown in the fictional example in the figure 2.2. We can see that in the context of three verbs *eat, write* and *read*, the words *article* and *book* are closer together while *rice* is positioned further from them in this three dimensional semantic space.

These vectors are called word embeddings because words are embedded as vectors in a multidimensional space (Jurafsky and Martin, 2018). They are also called *distributional semantic models* or *distributed word representations*. Besides being able to capture semantic similarity (*couch* vs. *sofa*, *article* vs. *book*) and semantic relatedness (*tea* vs. *cup*, *house* vs. *kitchen*), these models are also able to capture other types of semantic, syntactic and other relationships between words, such as gender, verb forms, relations between cities and countries, etc. This was firstly pointed out by Mikolov, Chen, et al. (2013) and in the figure 2.3 we show a simplified example of how these models are able to capture simple linear relations.

Distributed feature representation has now become a standard way of representing features for NLP tasks and it has shown state-of-the-art results in many of them. The available distributional models will differ, however, based on the way they

represent the words' context - word vectors can be learned from a context as wide as a document or as narrow as the words' nearest neighbours in a sentence (Jurafsky and Martin, 2018). When learning the distributions, we decide on the size of a context window and, depending on it, we can learn more syntactic or more semantic meaning representation of a given word. If the content window is narrow, we will manage to capture more of the syntactic meaning, while if we take a wider context, we will capture more of its semantic meaning.

The vector models have shown to be very useful for representing word meaning and there are various models available that are commonly used in NLP research for this purpose. These models are normally based on co-occurrence matrices, that is, how often words co-occur. There are two main types of co-occurrence matrices: term-document matrix and term-term (term-context) matrix (Jurafsky and Martin, 2018). The former contains the number of times each word appeared in each document, which means that every document is represented by a vector of word counts. In the latter model, each word of a vocabulary is represented as a vector of co-occurrence counts, i.e. how many times the target word appears in the context of another word (or a context window). In a lot of cases, the length of a word vector tends to be the size of the vocabulary, which can be anywhere between 10.000 and 50.000 words (Jurafsky and Martin, 2018).

In both of these types of matrices, not all the words will co-occur with every document or every context word in the vocabulary. Therefore, a lot of numbers in a vector will be zeros. These kinds of vectors which tend to have a lot of zeros are called *sparse*. They are not always easy to process because they tend to be quite large and due to the fact that the zeros do not carry any truly significant information about the word meaning.

Instead of using raw frequency to represent the meaning of words or documents, there are approaches that were shown to work well, which introduce weighting of the raw counts in order to give more importance to certain contexts or words over others. For example, there are particular words, such as *a, the, for, in*, etc. (called function words), that will appear across a lot of different documents and are, therefore, not good indicators of the document's meaning. The goal of a weighting approach would be to give less weight to these words in a certain document retrieval task as they tend to be less informative. There are many approaches in NLP that introduce some type of a re-weighting on the raw vectors, but the two very common are Tf-idf and Positive Pointwise Mutual Information (PMI) models.

All of the previous approaches deal with sparse vector representations which, how-

ever, have been shown to work less efficiently than their alternative - dense vectors, which consist of numbers that are mostly non-zeros. (Jurafsky and Martin, 2018). Since the most recent research in NLP has almost completely switched to utilizing these vector representations as features, the term *word embeddings* is now mostly used to refer to dense word vectors. One of the main differences between word embeddings and DSMs is that word embeddings are learnt by predicting the words (either predicting a word based on a context or predicting the context based on the given word), while DSMs were learnt, as we have seen before, by taking into account word co-occurrence. This was also proven by Baroni et al. (2014) who pointed out that word embeddings models performed better than DSMs in a range of NLP tasks, such as semantic relatedness, synonym detection, concept categorization, selectional preference and analogy task. This could be due to several factors - firstly, with dense vectors, classifiers trained for a particular task have to learn less weights for representing word meaning than with sparse, where classifiers have to learn several thousands of weights for every sparse dimension, and secondly, given that there are less parameters to learn, classifiers trained with dense vectors tend to generalize better and have less chances for over-fitting (Jurafsky and Martin, 2018).

The following are the most commonly used algorithms to learn dense vector representations:

- CBOW (Continuous Bag-of-words): The model was introduced by Mikolov, Chen, et al. (2013) and it predicts a target word based on a context window, which represents the sum of the vector representations of the words in it.
- Skip-gram: Another model introduced by Mikolov, Sutskever, et al. (2013) which predicts a context window based on a given word.
- GloVe (Global Vectors for Word Representation): This method was suggested by Pennington et al. (2014) and it is based on ratios of word co-occurrence probabilities (as opposed to DSMs that are based on co-occurrence probabilities only) that are computed using a weighted least squares objective in order to decrease the difference between the dot product of the vectors of two words and the logarithm of the number of their co-occurrences (Ruder et al., 2017).
- FastText: It was introduced by Bojanowski et al. (2017) as an extension of the Skip-gram model by incorporating a way to account for out-of-vocabulary words. This was done by including character n-grams, where the words that were not seen in the training data are represented by the sum of the vectors of their character n-grams (sub word units).

- ELMo (Deep Contextualized Word Representations): This model is currently state-of-the-art in word embeddings and it was introduced by Peters et al. (2018). Similarly to FastText, ELMo embeddings are character based, which means they represent words as a combination of their sub-units and like that have representations even for OOV words. On the other hand, as opposed to all other models, the word presentations depend on the whole context they are used in. ELMo embeddings are, in fact, concatenations of the activations on several layers of the deep bidirectional language model (biLM), since different layers encode different types of information about a word (Peters et al., 2018). For example, lower level layers predict well parts-of-speech, while higher levels tend to encode better word-sense disambiguation. Using these embeddings in several NLP tasks has already proven very successful and lead to increased accuracy of the models for semantic parsing, question answering, textual entailment and sentiment analysis (Peters et al., 2018).

2.2.1 Cross-lingual Word Embeddings

The emergence of the previously described approaches to learning monolingual word embeddings, gave rise to several approaches proposing to expand their ideas to a cross-lingual scenario and enable knowledge transfer between different languages for multilingual NLP tasks. This would potentially allow us to train models simultaneously for different languages, as well as create a possibility to transfer knowledge between resource-poor and resource-rich languages (Ruder et al., 2017).

There are multiple models and algorithms proposed to project word embeddings from different languages to a common vector space. They differ in the approach they take, but could be mainly split in those that are based on word-aligned, sentence-aligned, document-aligned data, lexicons and those that do not use any parallel data (Ruder et al., 2017). For the purpose of this thesis, we will be using word-aligned approaches, that is, the approaches that align words in different languages to a common vector space by using bilingual or cross-lingual dictionaries with word pair translations. We will focus mostly on these as they are the most dominant and most successful in the literature currently. These approaches can be further split as follows:

- Monolingual mapping: These types of methods make a great use of monolingual word embeddings, which are first independently trained on large corpora of text. After these were obtained, this group of approaches tries to project the monolingual embeddings to a common space, which is normally

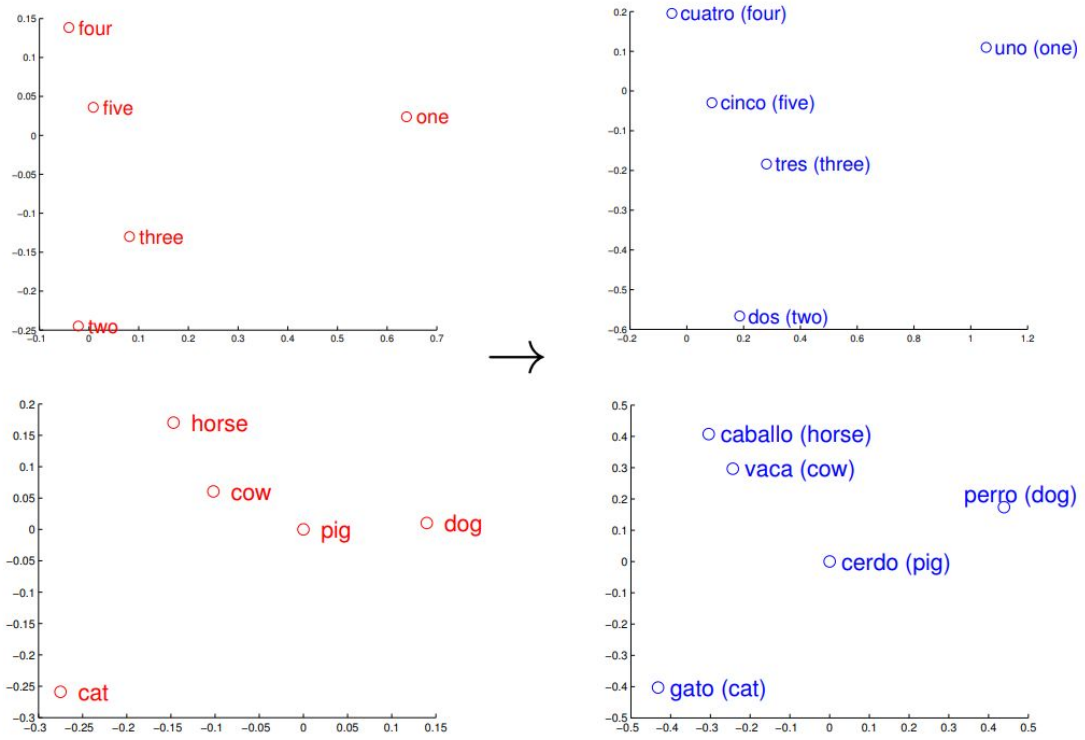


Figure 2.4: Distributed word representations of numbers (up) and animals (down) in English (left) and Spanish (right). (Mikolov, Le, et al., 2013)

done by learning a transformation matrix that would allow us to map word representations in one language to the representations of the other (Ruder et al., 2017). This method could be well described with the observation made by Mikolov, Le, et al. (2013) who noticed that geometric relationships between words tend to be similar in different languages. As shown in the figure 2.4, the words for numbers and animals in English and Spanish have a similar geometric shape in a vector space. This observation then gave rise to the idea that we could learn a linear mapping from one vector space to another by leveraging bilingual dictionaries. This mapping is learned by minimizing the mean squared error (MSE) between bilingual word pairs. Since this work, there has been a tendency to try to use as little resources as possible to obtain the linear mappings. The most recent work of Conneau et al. (2018) has suggested an unsupervised method for learning a linear projection based on a discriminative adversarial objective.

- Pseudo-cross-lingual approach: This approach trains multilingual embeddings using monolingual word embedding tools that are used and trained on an automatically constructed pseudo-cross-lingual corpus, which would contain text from both the source and target languages (Ruder et al., 2017). The expected result here is for the model to be able to learn cross-lingual

relations by being exposed to this type of cross-lingual context.

- Joint optimization: This class of approaches train their models on both parallel and monolingual data, while jointly optimizing a combination of monolingual and cross-lingual losses (Ruder et al., 2017).

There are also models that include methods which train crosslingual embeddings using comparable, instead of parallel data, such as language grounding models and comparable feature models. However, since their methods are not the ones we are going to use in this thesis, we will not cover them. For further information about them, however, we refer the reader to Ruder et al. (2017).

Despite the fact that cross-lingual embeddings present the current state-of the-art in multilingual and crosslingual NLP and are a great resource to further influence its development, they have, however, some disadvantages. We have outlined a couple of them here (Ruder et al., 2017):

- Word order: Given that most of the approaches are based on the bag-of-words representation, the models we currently have are not able to capture word order. And, as we will see later, this could be one of the potential sources of error our semantic role labeling model has, since these kind of models would tend to assign the same representation to, for example, a sentence in active and passive voice.

- Polysemy: The problem here arises because of the fact that a monosemous word in one language can be polysemous in another, and the models are currently not able to capture this feature of the target word through a simple linear alignment method. What happens usually is that we would align the monosemous source word to one of the senses of the target word, which is not necessarily the correct one.

- Creating a shared feature representation seems to work fairly well for closely related languages. The further the linguistic structures of languages are apart, the more difficult it is to create a shared representation that would capture all of their linguistic particularities. On top of that, if the languages are too different, there is a chance of not being able to transfer any knowledge between languages and have a negative transfer (Ruder et al., 2017).

- And finally, different cross-lingual embeddings are evaluated on different tasks and under different conditions. If they functioned well in one, it does not mean they will function equally well in another task. Besides, it was shown that for different tasks, embedding models require different forms of supervisions. An interesting observation, and perhaps useful for our re-

search, is that for the task of parsing word-level alignment models tend to perform better as they are able to capture syntactic information more precisely (Upadhyay et al., 2016).

In the next two chapters, we will show how multilingual embeddings could be used for the task of shallow semantic parsing in a low resource setting, where either little or no annotated data is available. The embeddings we will use fall under a monolingual mapping category. We also try to compare two types of pretrained cross-lingual embeddings that were integrated into our SRL system and allowed for a knowledge transfer between a resource-rich language and languages with a smaller amount of resources available. We will cover in more detail in chapter 4 the two types of cross-lingual embedding alignment approaches we will use in this thesis.

2.3 Neural Network Approaches to NLP

Until recently, most of the machine learning approaches used in the field of NLP were linear models (support vector machines, linear regression, etc.) that were using as input high-dimensional and sparse vectors as features. However, influenced by the use of NNs in other fields, such as computer vision and pattern recognition, NLP slowly started turning to non-linear, neural network models trained on dense vectors that are able to automatically learn feature representations (Goldberg, 2016; Young et al., 2018). This change of direction came along with advancement in word embedding learning models as well as deep learning models, which resulted in state-of-the-art performances on various NLP tasks: language modelling (Yang et al., 2017; Krause et al., 2018), POS-tagging (Ling et al., 2015; Bohnet et al., 2018), dependency parsing (Dozat and Manning, 2016; Clark et al., 2018), named entity recognition (Akbik, Blythe, et al., 2018) and machine translation (Vaswani et al., 2017; Edunov et al., 2018), among others.

In this section, we will give an overview of the common neural architecture and cover in more detail one of the methods commonly used to tackle challenging NLP tasks - Recurrent Neural Networks (RNNs) and one of their modifications - Long Short-Term Memory network (LSTM). This is, in fact, the method that we will make use of for the task of semantic role labelling in this project.

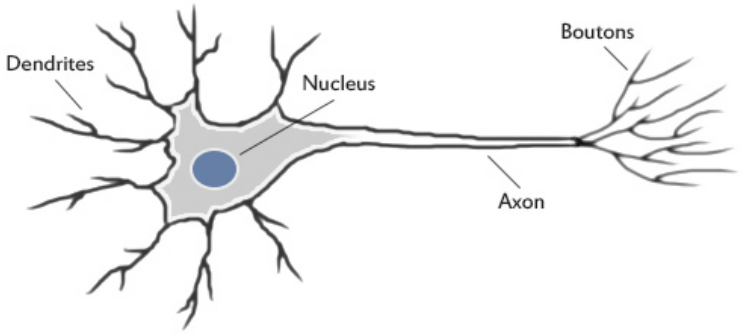


Figure 2.5: Illustration of a biological neuron.(Jacobson, 2013)

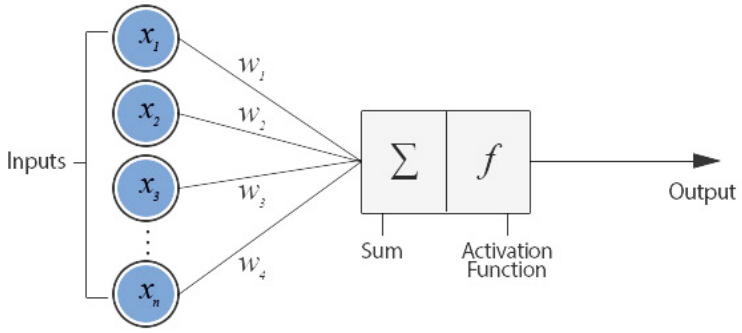


Figure 2.6: An example of a single-layer percepteron.(Jacobson, 2013)

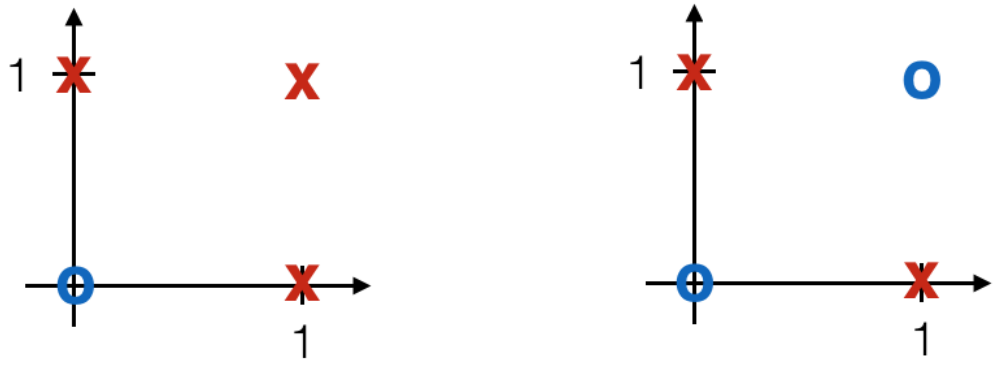


Figure 2.7: Left: linearly separable data points. Right: not linearly separable data points.

2.3.1 Common Architecture

Neural Network models represent a class of machine learning models that attempt to process information similarly to neural networks in biological brains, only on a simplified scale. In figure 2.5, we showed an example of a neuron: neurons receive input information through structures called *dendrites* and send an electrical signal further via axons all the way to boutons in the network if the sum of impulses received from dendrites is higher than a certain threshold (Jacobson, 2013). Similarly, artificial neural networks consist of neurons as their basic units, also called *nodes* or *perceptrons*. In the figure 2.6, we can see the similarity between a perceptron and a biological neuron - a perceptron also receives several inputs $(x_1, x_2, ...x_n)$ and is supposed to produce an output. All the inputs received by a perceptron are separately weighted $(w_1, w_2, ...w_3)$ depending on their relative importance when compared to each other. Weights can be either positive or negative. The node then sums the weighted input and if that input is over a certain threshold, it produces an output - just like we have seen with the biological neuron. Whether the model will or will not *fire* or produce an output is defined by an *activation function* (for example, tanh or ReLU functions), which is used to convert the input into a more meaningful output (Jacobson, 2013). This basically means that when we train a model for a certain task, we want the perceptron to fire whenever the model manages to learn a new pattern from the data we are using, which is modeled with an activation function. By changing the weights and the threshold, we can get different decisions from the system. And finally, a perceptron can have one additional input called *bias* which serves to shift the activation function either to the left or to the right.

A perceptron is, however, a linear classifier and it cannot deal with non-linearities. In the figure 2.7 on the left, we show an example of data points that can be separated by a single line, while on the right, we have data that is not possible to separate by one unique line. If we have data that is not linearly separable, we will need a model that has a higher representational capacity - a neural network. (Goldberg, 2016)

A neural network would normally consist of a network of such perceptrons. In the figure 2.8, we have a network consisting of an input layer with four input nodes, one output layer with two nodes and one *hidden layer*. Each input node is used as an input to each node of the hidden layer, each of which have, thus, several inputs and produce a single output fed to the output layer (Goldberg, 2016). Given that there is only one direction in which information could move to in this network, it is called a *multi-layer perceptron* (MLP) or a *feed-forward network*, since the

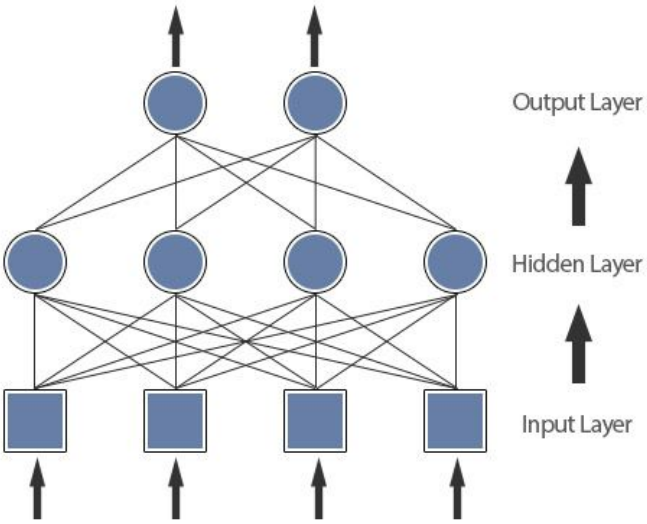


Figure 2.8: Feed-forward neural network (bias not shown). (Jacobson, 2013)

information can only move forward through the network. In other words, output from one layer is used as an input to the following layer.

2.3.2 Recurrent Neural Networks (RNNs)

In a feed-forward network, we made the assumption that all inputs are independent of one another. However, that is not always the case, especially when we deal with text as input. For example, if we wanted to predict the next word in a sentence based on the previous word, we could implement another type of network that would allow us to do this in an easier way than the FFN could. In this case, we could use a *Recurrent Neural Network* (RNN), which does the same computation on each of the elements in a sequence (thus the name) and it assumes that the output of the network is based on all of the previous calculations (Olah, 2015).

If we look at figure 2.9 on the far left, we can see that a simple RNN receives an input x_t and has an output h_t, with the arrow denoting a loop. If we *unfold* this network (figure 2.9 on the right), we will see that the RNN is nothing else then a sequence of connected FNNs that are following one another in continuous time step (Goldberg, 2016). Each FNN is passing its output information to the next one in the sequence. In other words, each FNN receives two inputs at a time - one from the current time step and one from the hidden layer of the previous time step.

Given their ability to model sequences and predict the next element based on the previous ones, they are successfully applied to various NLP tasks, such as language modeling, POS-tagging, machine translation, etc. (Olah, 2015). In

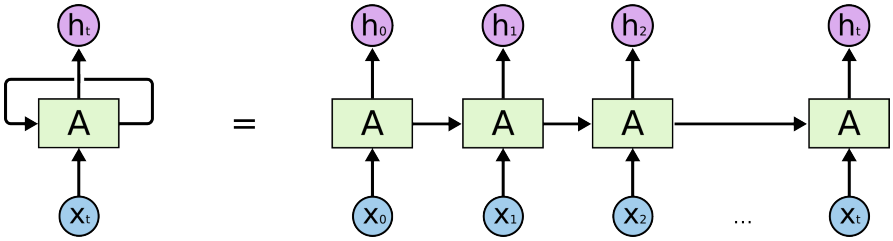


Figure 2.9: An unfolded recurrent neural network. (Olah, 2015)

particular, a special type of RNNs is used successfully in these tasks, which is called *Long Short-Term Memory* network (LSTM). This network had a big success in NLP mainly because it is able to tackle the main issue the typical RNN has - long-term dependencies. RNNs were shown to be very biased towards the most recent input and are, thus, limited in terms of how much of previous information they can keep (Goldberg, 2016). This is also related to the problem of *vanishing gradients* or their counterpart *exploding gradients* (Pascanu et al., 2014). The issue arises from the fact that neural network parameters are optimized using error backpropagation which calculates gradients of the loss function through the chain rule (Zilly et al., 2017). Activation functions that are commonly used (such as tanh) have derivatives between -1 and 1, so the multiplication sequence of such small numbers will result in the gradients to approach zero with each time step the network goes through. Similarly, the gradients can *explode* if the weights of large numbers are multiplied (Pascanu et al., 2014).

2.3.2.1 Long Short-Term Memory Network (LSTM)

In order to deal with the issue of vanishing gradients and long-term dependencies, Hochreiter and Schmidhuber (1997) proposed a version of RNN, LSTM, that have memory cells instead of the nodes in the hidden layer. We show the difference between an RNN and LSTM network in the figures 2.10 and 2.11, respectively. While RNNs have several repeating modules of neural networks containing only one tanh layer, LSTMs contain a four layer network (Olah, 2015).

Following the figure from left to right, the structure of the LSTM is as follows. Firstly, in the sigmoid layer a decision is made about whether information from the input is going to be kept in the cell C_t. The sigmoid layer is called *forget gate* and it takes into account the output of the hidden layer of the previous time step h_{t-1} and the input from the current time step x_t (Goldberg, 2016). The decision is binary - it outputs 1 if the information is relevant and should be kept, and 0 if it should not for all the numbers from the previous cell C_{t-1}. It is formally written

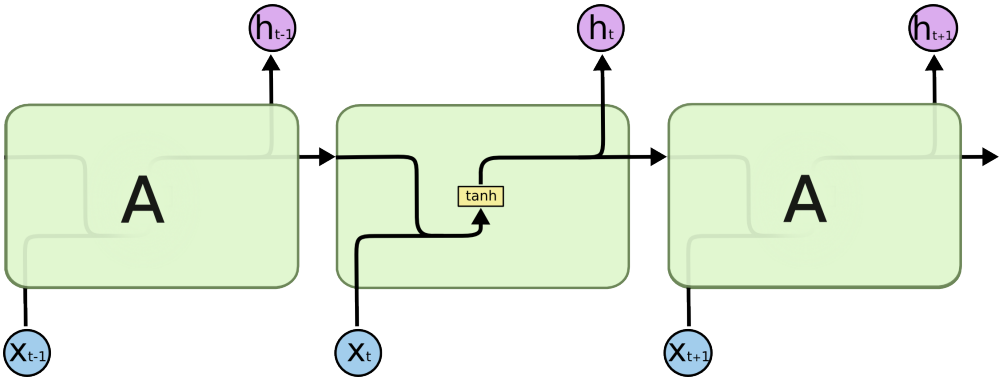


Figure 2.10: A standard RNN structure. (Olah, 2015)

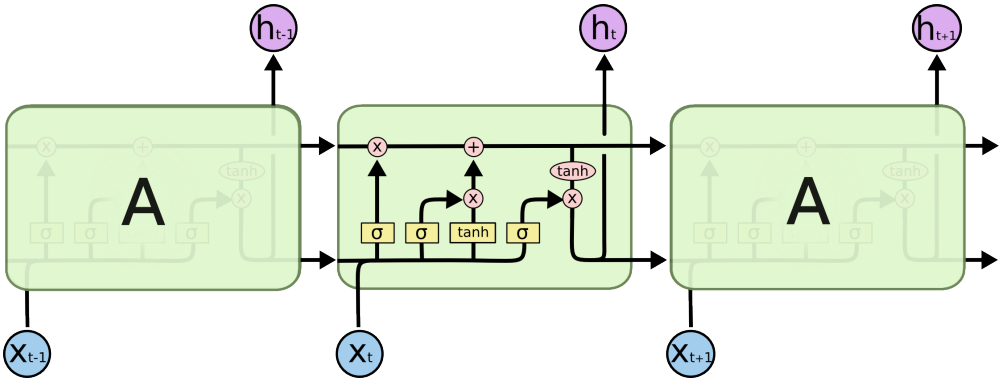


Figure 2.11: A structure of a LSTM network. (Olah, 2015)

as follows, where f_t is the output of the forget gate:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Next, it needs to decide what new information should remain in the current cell state. First, the next gate, *input gate* (i_t), modulates which values should be updated, while the tanh layer is used to create vectors of new potential values for the update, C'_t. These two steps are then multiplied to obtain an update to the state (Olah, 2015). Again, these are formally calculated as:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$C'_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C)$$

Then, the previous cell state, C_{t-1} , needs to be updated with the new values and create a new state C_t. In order to do that, the previous state is multiplied by the output of the forget gate, we add the multiplication of the new values and the

output of the input gate:

$$C_t = f_t * C_{t-1} + i_t * C'_t$$

And finally, the last step determines what the output, o_t, of the cell will be, which is decided in two steps. First, the sigmoid layer filters what part of the cell state will be outputted and second, the cell state is ran through a tanh layer in order to set the output between -1 and 1. The results of the sigmoid and the tanh function are then multiplied to create the output vector h_t (Goldberg, 2016, Olah, 2015):

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * tanh(C_t)$$

In this thesis we will make use of a variation of an LSTM network called a *bi-directional LSTM* (bi-LSTM), which is able to process sequences of words in both directions: from x_1 to x_t and from x_t to x_1 by concatenating outputs that are produced in both directions (Schuster and Paliwal, 1997). This type of network has proven very successful in recent SRL research and has produced a range of state-of-the-art results, which are going to be described in the next section.

Chapter 3

Related Work

After having discussed the background material discussed for this thesis in the previous chapter, we now look in the most relevant and related work from the literature. What is meant by this is that we will not go through the annotation projection and unsupervised methods which are often applied in cross-lingual scenarios, because their approaches differ significantly from ours. On the one hand, annotation projection relies heavily on parallel data and other resources such as POS-tags, syntactic parsing output, etc., while on the other, unsupervised methods have a very different approach from ours and tend to perform with a significantly lower accuracy than both annotation projection and model transfer systems. Even though we are not using a supervised method in a traditional way, we are, however, making use of a modified state-of-the-art supervised model to apply it to new languages. Therefore, this section will firstly give an overview of the current research on supervised methods which typically do not require using any syntactic data, as we consider high quality syntactic trees to be a very high resource annotation which is not available for many languages. Then, we will give an overview of the model transfer methods used in SRL until now and, in the end, we will look at the most recent examples in the field of dependency parsing transfer as some of the approaches could be applied in an SRL scenario as well, since the tasks are related.

3.1 Syntax-agnostic Neural SRL Methods

As we have shown in section 2, in the past, most of the SRL systems tended to use syntactic information as one of the important features of both traditional and neural machine learning models. However, a lot of recent work seems to turn this

story around, proving that syntactic parsing is not a necessary step for successful semantic role labeling. One of the first neural models in SRL by Collobert et al. (2011) was actually the first model to make an attempt of doing semantic parsing without syntactic features. However, this model was not successful as it did not manage to perform better than the traditional models at the time. The first successful syntax-agnostic system was the one of Zhou and Xu (2015) who constructed a deep bi-directional long short-term memory (bi-LSTM) network with a conditional random field (CRF) model at the top for predicting the output tag sequence. The model achieved an F1 score of 81.27% on the CoNLL-2012 shared task data. This was a big turn in SRL research since this model did not rely on any syntactic information, which was difficult and expensive to obtain and which was causing the biggest amount of errors in SRL systems until that time (as pointed out by Pradhan, Ward, et al. (2005)). On top of this, their work laid ground for many of the upcoming state-of-the-art systems, which not only applied a similar neural architecture, but also avoided using syntactic features.

Following this work and building upon it, He, Lee, Lewis, et al. (2017) proposed a system that, as opposed to the work of Zhou and Xu (2015), uses a deep highway bidirectional LSTMs with constrained decoding. This system became the new state-of-the-art with an F1 score of 81.7% on the CoNLL-2012 test set. Given its competitive performance, this system was a very good proof that LSTMs have a great potential to learn underlying syntactic structure of sentences even without using any syntactic features. Their model is the one we will make use of in this thesis for the purpose of SRL model transfer, since it was the state-of-the-art at the time we started our progress on this work. A detailed overview of the system will, thus, be given in the following section (section 4) which describes our approach.

A slight change to this model was suggested a couple of months later which was following the new advancements in the field of word embeddings - the appearance of already described ELMO embeddings by Peters et al. (2018). Changing the input to the model from pretrained GloVe to ELMO embeddings, significantly improved the accuracy of the model, from 81.7 to 84.6 F1. This is due to the fact that ELMO manages to capture context-dependent characteristics of word meaning as well as words' syntactic information (Peters et al., 2018), which is especially beneficial for this task.

Up until this point, it was shown that the two biggest issues in syntax-agnostic SRL were structural information on the one hand and long-range dependencies on the other. The system that attempted to tackle these was by Tan et al. (2018) and it involved using a deep attentional neural network (DEEPATT) which relies on

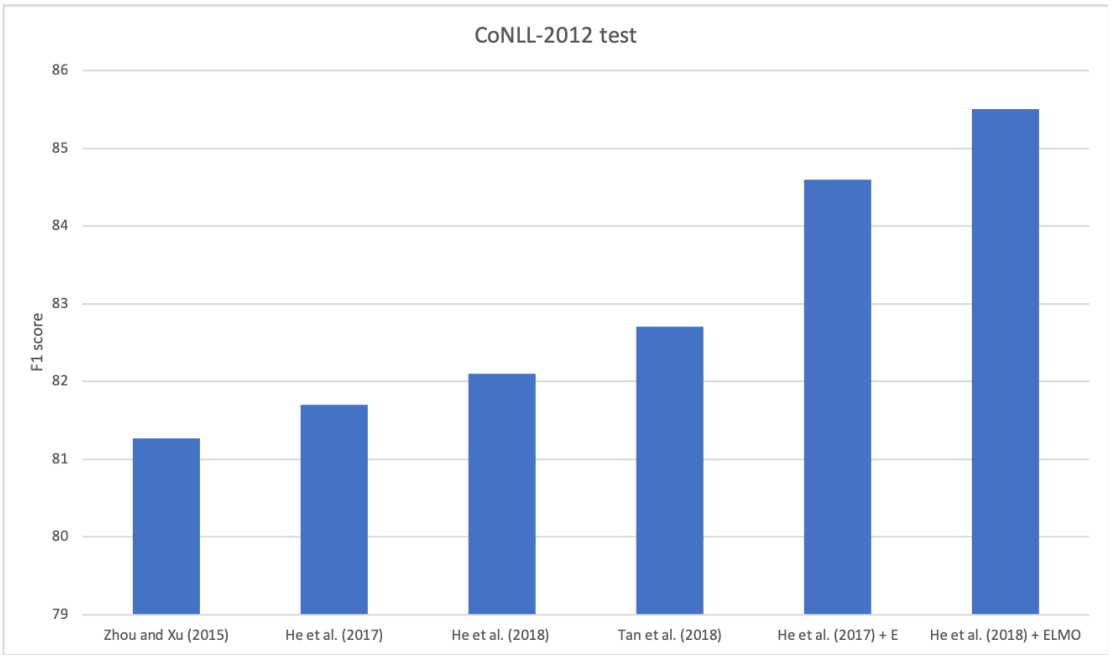


Figure 3.1: Overview of syntax-agnostic systems tested on the CoNLL-2012 test data.

self-attention mechanisms to directly draw the global dependencies of the inputs. This model can, as opposed to bi-LSTM models, create connections between two arbitrary tokens in a sentence and thus allow for distant elements to interact with one another by shorter paths, making it possible to deal with long-range dependencies. The model achieved 82.7 F1 score on the CoNLL-2012 shared task dataset which became a new state-of-the art after He, Lee, Lewis, et al. (2017).

And finally, the current state-of-the-art system is again proposed by He, Lee, Levy, et al. (2018) and with the use of ELMO word embeddings. The difference from the 2017 model is that instead of assuming predicates as given, the model tries to predict them. This is the biggest contribution of this system as it is the first span-based SRL model to try to do this. The very idea actually came from a coreference model by Lee, He, et al. (2017) and given that a similar model worked fairly well on an SRL task, gives place to the conclusion that a similar approach could be used in other tasks that use span-based labelling, such as syntactic parsing and relation extraction. On top of this, the model shows, similarly to Tan et al. (2018), that it is able to tackle long-range dependencies as well. Finally, the system achieved an F1 score of 85.5% on the CoNLL-2012 dataset.

In order to have a better overview of all the previously described systems, we have showed their results in the figure 3.1 bellow. The results presented are the ones on the CoNLL-2012 data as this is the dataset that we will use for training our models (more about this in the next section).

3.2 Model Transfer of SRL

To the best of our knowledge, the only model transfer method applied to the task of SRL by using distributed feature representation was the one proposed by Kozhevnikov and Titov (2013). As opposed to the previous work that mostly made use of delexicalized parsers to reduce the lexical gap between different languages, Kozhevnikov and Titov (2013) use a shared feature representation for the languages involved - English, French, Czech and Chinese. They trained models for the following language pairs: EN-ZH, ZH-EN, EN-CZ, CZ-EN and EN-FR (where ZH is Chinese, CZ Czech, En English and FR French). In other words, every language was used both as a source and as a target, except in the case of French which was used only as a target language.

In order to build the transfer model, they used the following features: (1) a shared feature representation, which included using cross-lingual word clusters and cross-lingual distributed word features, (2) unlabeled dependencies or, if non-existent for a certain language, syntactic information was transferred from the source to the target language, (3) universal POS-tags, (4) glossed word forms. In order to train their mdoels, they used a linear classifier by Björkelund et al. (2009), which is comprised of a set of linear classifiers.

Compared to a cross-lingual projection systems they used as a baseline, the model performed competitively given that no direct parallel data was needed. For the pairs EN-ZH, ZH-EN, EN-CZ, it even outperformed the projection model. The results obtained on the CoNLL-2009 dataset are given in the figure 3.2. On the left are the F1 scores for the task of argument identification, while on the right are the results for argument classification. In both of the cases, Kozhevnikov and Titov (2013)'s results are compared to their annotation projection system. The results vary widely depending on the both source and the target languages in question, as well as on the type of syntactic information that was used (original or transferred).

The authors showed a successfully attempt of incorporating lexical features into the transfer model, which is the main achievement of their approach. However, this method requires a lot of resources expensive to obtain. The authors claim that this model could be used in a low-resource scenario, but it requires dependency parses in order to achieve good results, which is expensive information to obtain. In order to try and overcome this issue, they transfer syntactic information from the source to the target language. The transferred parses introduce a lot of chances for errors since, by making the transfer, we are trying to fit one language structure

Setup	Syntax	TRANS	PROJ
EN-ZH	trans	34.5	13.9
ZH-EN	trans	32.6	15.6
EN-CZ	trans	46.3	12.4
CZ-EN	trans	42.3	22.2
EN-FR	trans	61.6	43.5
EN-ZH	orig	51.7	19.6
ZH-EN	orig	53.2	29.7
EN-CZ	orig	63.9	59.3
CZ-EN	orig	67.3	60.9
EN-FR	orig	71.0	51.3

Setup	Syntax	TRANS	PROJ
EN-ZH	trans	**70.1**	69.2
ZH-EN	trans	**65.6**	61.3
EN-CZ	trans	**50.1**	46.3
CZ-EN	trans	53.3	**54.7**
EN-FR	trans	65.1	**66.1**
EN-ZH	orig	**71.7**	69.7
ZH-EN	orig	**66.1**	64.4
EN-CZ	orig	**59.0**	53.2
CZ-EN	orig	61.0	60.8
EN-FR	orig	63.0	**68.0**

Figure 3.2: Results reported by Kozhevnikov and Titov (2013). Left: argument identification (F1). Right: argument classification (accuracy)

into the another, which is very prone to errors (Kozhevnikov and Titov, 2013).

Since we are making a comparison to this work in our project, we are implementing a different approach. We try to keep our model as low resource as possible and we do not use any syntactic information. On the other hand, we try to experiment with using no target language(s) data or using only a small amount of target data (100 sentences). By exposing the model to the target data, it should implicitly learn the syntax of the target language. This was already attempted by J. Guo et al. (2016) for the task of model transfer for dependency parsing and it showed promising results, as it will be pointed out in the next segment.

3.3 Neural Model Transfer of Syntactic Dependencies

Given that the only system for SRL model transfer was the one of Kozhevnikov and Titov (2013, 2014) who use a traditional machine learning model, we will shortly look into some of the related research in the field of dependency parsing that makes use of a neural approach in combination with different types of distributed word representations. These models of dependency parsing are normally referred to in the literature as *cross-lingual representation learning methods* because at the time they were published, the biggest focus was on finding the best way to learn cross-lingual word embeddings and apply them to the task of model transfer. Here we will cover only the methods that could be applied to a lower resource setting - there are other approaches that give good results as well, however they include using annotated treebanks (for example, Duong et al. (2015a)), which is

the approach very distinct from ours.

One of the first neural models for the transfer of a dependency parser was proposed by Zou et al. (2013) and Xiao and Y. Guo (2014). Similarly to Kozhevnikov and Titov (2013), they make use of word embeddings to reduce the lexical feature gap in a cross-lingual scenario, but as opposed to them, these two approaches use them as augmenting features to train a delexicalized parser on a resource rich source language and apply it to another. They train their own dense vectors using a deep neural network and bilingual dictionaries from Wiktionary. Difference is, however, that Zou et al. (2013) first trained monolingual embeddings for each language and then tried to transform the representations from one language to another using machine translation alignments. On the other hand, Xiao and Y. Guo (2014) jointly trained cross-lingual embeddings for the two languages involved by using a small set of bilingual pairs of words from Wiktionary, a dictionary which is freely available. They implemented the learnt embeddings to the MSTParser (McDonald et al., 2005) which they trained first on English and then applied to the other language. Finally, their results showed a significant improvement over the baseline delexicalized parser, as well as that the distributed lexical features combined with the structures learned by a delexicalized parser could be very useful for reducing the amount of errors in model transfer approaches.

Following this work, there is interesting approach by Søgaard et al. (2015), who suggested a parsing model that incorporates embeddings learned from multiple sources via inverted indexing in Wikipedia. They applied the method on various NLP tasks besides dependency parsing and even though they concluded that their embeddings did not improve much the results from previous work, it did however point to a promising future direction for learning cross-lingual word embeddings.

More recent work by J. Guo et al. (2016) builds upon the previous work, but tries to implement a different approach for using cross-lingual embeddings - by mapping vocabularies from different languages to a common vector space. They also noticed that they can additionally reduce the error rate by embedding cross-lingual clusters (similar to Täckström et al. (2012)). And finally, they have also shown that even a small amount of target labelled data (100 sentences) can significantly improve the parsing results because it allows for parameter adaptation.

3.4 Conclusion

All of the described approaches differ from one another mainly based on the approach they use to learn cross-lingual embeddings. However, what they have in

common is that besides using the embeddings themselves to reduce the lexical gap between languages, they also incorporate another approach to learn the non-lexical features in a target language - usually, word clusters. That being said, in our work we will try to investigate if our neural cross-lingual model is able to automatically learn clusters in these languages by incorporating state-of-the-art multilingual embeddings aligned in the same vector space which should allow for the model to access both their lexical and non-lexical features at the same time.

Chapter 4

Cross-lingual Model Transfer for SRL: Design and Methodology

The proposed thesis aims to use the deep SRL model of He, Lee, Lewis, et al. (2017) in order to build new SRL parsers for French and Chinese. We attempt to do this by training the model on the English training data while using pretrained cross-lingual word embeddings (Smith et al., 2017) to reduce the lexical gap between these languages. This also makes it possible for the system trained solely on English to be directly applied to French and Chinese sentences. Additionally, this system is particularly useful for the purpose of our task of SRL labeling as it does not use any syntactic information, which was shown to be beneficial for the languages with significantly different syntactic structures. This is true for the languages we are using for this project as they belong to different language families: Germanic, Romance and Sino-Tibetan.

In this chapter, we will firstly give a description of the model that we will implement, followed by the embeddings that we will use for its initialization in the conducted experiments. Then, we will give an overview of the data we trained and tested the system on. And finally, we will present the implementation setup used throughout all the experiments as well as the method used to evaluate the performance of our systems.

4.1 Model Description

The model we will use for this project is, as already mentioned, the one proposed by He, Lee, Lewis, et al. (2017) with certain modifications that will be explained in the upcoming sections. This model proved very successful for the task of SRL

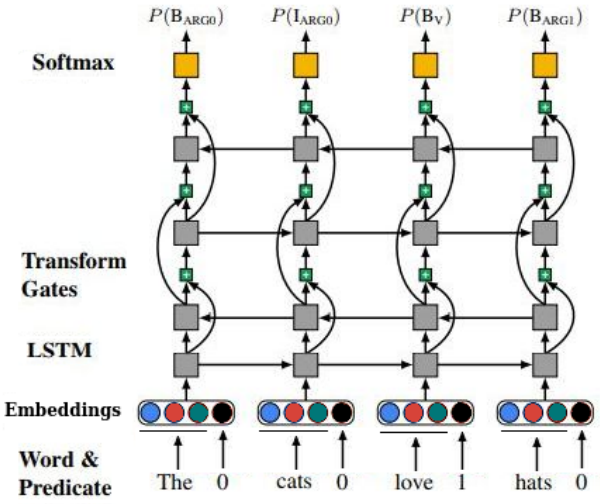


Figure 4.1: Example of the highway bi-LSTM with four layers. It takes a sequence of word-predicate pairs as input and produces a sequence of BIO tags as output. The curved connections stand for highway connections, while the plus symbols stand for transform gates which are controlling inter-layer information flow. (He, Lee, Lewis, et al., 2017).

for the English language, which could be mainly attributed to the fact that it incorporates the latest advancements in the field of deep learning - the authors implement a bidirectional LSTM to tackle this task, however, they simplify the input and the output layer, they incorporate highway connections (Srivastava et al., 2015), use recurrent dropout (Gal and Ghahramani, 2016) as well as pose certain constraints to the output sequence of labels by using A* decoding algorithm (Lewis and Steedman, 2014). We will take a more detailed look at each of these below.

The structure of the model is presented in the figure 4.1, where it can be seen that what it essentially does is that given a sequence of word-predicate pairs, it predicts a sequence of outputs. Each label of the output is part of the BIO tagset. In this type of annotation, the label *B-tag* denotes the beginning of an argument span marked with a certain label, *I-tag* denotes an inside element of a tagged argument span, while *O* is used to mark all the unlabelled tokens, that is, those tokens that are outside of a tagged argument span[1]. The model finds and outputs the highest-scoring tag sequence over the span of all possible tag sequences.

Furthermore, the model represents a stacked (or deep) bi-LSTM network, which is the extension to a classical bi-LSTM model with the main difference being additional hidden layers (in our case eight), where each layer has several memory

[1]This is shown in more detail and followed with an example in the Section 4.3 where we introduce the data used for the project.

cells (Pascanu et al., 2014). Stacking LSTM hidden layers creates a deeper model, and the depth is what is usually thought of being the main contributor to the model's success (Pascanu et al., 2014; Hermans and Schrauwen, 2013). Each layer of the network processes one part of the task and then sends it to the next until it reaches the last layer which produces the output (softmax). On top of that, additional layers can process further what was learned in the previous layers, and it is believed that deeper layers in the network are able to create higher level of abstraction (Pascanu et al., 2014). For example, by working on speech recognition, Graves et al. (2013) found out that depth of a network was more important than the number of cells for obtaining better results. In fact, they showed that the error rate was constantly dropping as they increased the number of hidden layers.

It is clear that this model can benefit from using a deep neural structure, however, stacked LSTMs tend to have a similar problem with *vanishing* and *exploding gradients* as other RNNs (Pascanu et al., 2014). In Chapter 2, this problem was already introduced for a classic RNN architecture and it was said that LSTMs were presented as a solution for this problem. This was, in fact, the solution for the increased depth through *time*, which is proportional to the length of the input sequence. Here, the problem is now increased depth through *space* as we are stacking several LSTM layers on top of each other. In order to alleviate this issue deep networks have, He, Lee, Lewis, et al. (2017) incorporate *highway* connections proposed by Zilly et al. (2017). They essentially introduce another gatening mechanism to control the flow of information through the network - the *transform* gate is added in order to control how much of the activation is passed from one layer to the next (transform gates are marked with a plus sign in the figure 4.1).

And finally, in order to set constraints and ensure structural consistency between output tags, A* decoding algorithm over potential tags is used (suggested before by Lewis and Steedman (2014) and Lee, Lewis, et al. (2016)). These constraints can be split in BIO constraints and SRL constraints. The former ensures that the sequence of tags is a valid BIO transition - that I_{A0} is, for example, always preceded by a B_{A0}. The SRL constraints can be of two types: (1) Unique core roles (U), where each core role (A0-A5) can appear only once for each predicate; (2) Continuation roles (C), which refers to the fact that a continuation role (C-X) can appear only when its base role appeares before it (X). The authors experimented with using different types of syntactic decoding, but these were omitted from the final system as the only real improvement came from the BIO constraints.

4.2 Cross-lingual Word Embeddings

In this project two types of pretrained cross-lingual embeddings are used to initialize our models, the ones by Smith et al. (2017) for the main experiments and the ones by Conneau et al. (2018) for comparison purposes. As it has become practice in NLP recently, we use these embeddings in order to provide the model with an initial word-level representations. Using pretrained embeddings in itself can be seen as a type of transfer learning since we are leveraging information learned through training these embeddings on large corpora of text and applying it (i.e. transferring it) to a another NLP task such as SRL. Both types of approaches are using pretrained monolingual embeddings trained on Wikipedia corpus with the approach proposed by Bojanowski et al. (2017).

Most of our experiments are done using FastText embeddings aligned to a common vector space by using Smith et al. (2017)'s transformation matrices[2]. In order to obtain the matrices, the authors take 10000 most frequent words from the English FastText vocabulary and translate them to other languages through Google Translate API. This created dictionary is split into 5000 words for training and 5000 words for testing. In order to induce a shared bilingual space for a language pair, the authors use the created bilingual dictionary of translation pairs and the pretrained monolingual embeddings. If an English word has a phrase as its translation, they take the average of the word vectors that the phrase contains. The multilingual mapping is then done by minimizing mean squared errors between the translation pairs in the seed dictionaries, while to avoid the issue of *hubness*[3] they invert the softmax used to find word translations at test time and then normalize the probability over source words instead of target words (Ruder et al. (2017)). In order to preserve the original distances in the monolingual spaces, the authors apply orthogonal constraints to the linear transformations.

Besides these, we will make use of another type of cross-lingual embeddings that represent the current state-of-the-art - MUSE embeddings by Conneau et al. (2018). The authors make freely available pretrained cross-lingual embeddings for various languages, as well as high-quality bilingual dictionaries for aligning new cross-lingual embeddings[4]. In order to obtain the cross-lingual embeddings, the authors also use pretrained, monolingual FastText embeddings and then leverage

[2]The transformation matrices and the tutorial for obtaining the embedding alignments are available at https://github.com/Babylonpartners/fastText_multilingual

[3]Hubness refers to the problem in the approach to linear projection proposed by Mikolov, Le, et al. (2013) and it refers to the fact that some words tend to appear as nearest neighbours of a lot of other words in a vector space. This was first noticed by Lazaridou et al. (2015).

[4]The embeddings and the dictionaries are freely available here: https://github.com/facebookresearch/MUSE

bilingual dictionaries to learn the mappings between source and the target spaces. The main difference to the previous approach is that the authors use only two monolingual corpora and propose an adversarial learning approach to learn a linear mapping from source to target spaces. This approach involves having two systems where one (discriminator) tries to make a distinction between the mapped embeddings in two languages (tries to decide which language embeddings belong to), while the other (generator) tries to align the two spaces and like that prevent the discriminator of being successful. After the shared space is obtained, they extract the dictionaries created in this way and fine-tune the mapping with iterative Procrustes alignment by (Schönemann, 1966).

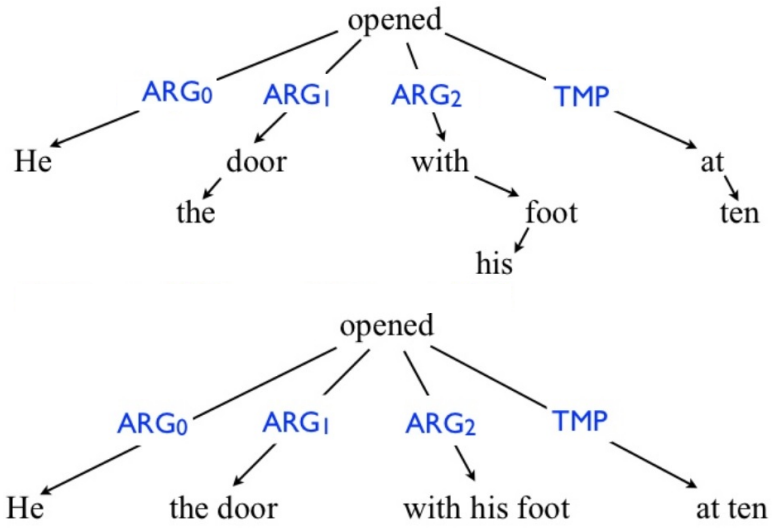


Figure 4.2: The difference between the span-based and dependency-based semantic role annotation (Choi and Palmer (2011)).

4.3 Data

The data we will use is comprised of three different datasets: one for the training and development of our models and two for their evaluation. For the former purpose we make use of the data from the CoNLL-2012 shared task (Pradhan, Moschitti, et al., 2012) for the English language, while for the latter we use the CoNLL-2009 shared task data (Hajič et al., 2009) for Chinese and the one from Van der Plas, Merlo, et al. (2011) for the French language. Even though the French data was obtained in a different way, it is in the same format as the Chinese data - CoNLL-2009 format (this is explained in more detail in the next section).

Before we present the datasets in more detail, there is an important piece of

information about the semantic role annotation in these datasets. Even though they are all annotated with PropBank style labels, there is a difference in how this annotation is applied across the datasets. While the CoNLL-2009 data is in the dependency-based format, the CoNLL-2012 data is in the span-based format. In the figure 4.2 on the previous page, the difference between the two formats is shown. While in the dependency-based format only the head of a phrase is labelled with a semantic role, in the span-based format the whole constituent is identified and then marked with BIO tag structure. In the BIO annotation, B presents the beginning of a tagged phrase, I denotes a word carrying a semantic role within that phrase and O means that the word is not carrying any semantic role label (it stands for *Outside* of a tagged phrase sequence). This type of annotation is illustrated by the example 6 below.

(6) [He$_{B-A0}$] opened [the$_{B-A1}$ door$_{I-A1}$] [with$_{B-A2}$ his$_{I-A2}$ foot$_{I-A2}$] [at$_{B-TMP}$ ten$_{I-TMP}$].

Samples of the two annotated sentences from the datasets are given in Appendix A, together with the explanation of contained information in each of the columns in the datasets.

The reason why we are obliged to use the data in different formats is that the original monolingual system of He, Lee, Lewis, et al. (2017) that we use for the implementation is constructed to use the data in the span-based format. On the other hand, the previous work we want to compare our results to (Kozhevnikov and Titov (2013) and Van der Plas, Merlo, et al. (2011)) is evaluating their systems on the CoNLL-2009 data which is in the dependency-based format. In order for our results to be comparable to theirs, we need to evaluate our systems on the same datasets.

4.3.1 Training and Development Data

The dataset we use for the training and development of our models is the English data from the CoNLL-2012 shared task, following the original experiment by He, Lee, Lewis, et al. (2017). The data was extracted from the news, broadcasts, talk shows, etc. and follows the PropBank for labelling predicate-argument relations (Pradhan, Moschitti, et al., 2012), which are in the span-based, BIO format explained earlier. Both verbal and nominal predicates are given in the task together with their sense disambiguation. Besides this information, the data includes named entities, correference information, POS-tags, syntactic parses, etc.,

but we do not use this information in our models. Instead, we only use word forms.

CoNLL-2012 English language data			
	Train	Dev	Total
Sentences	2.2M	300K	2.5M
Tokens	115K	15K	130K

Table 4.1: Number of sentences and tokens in the English training data.

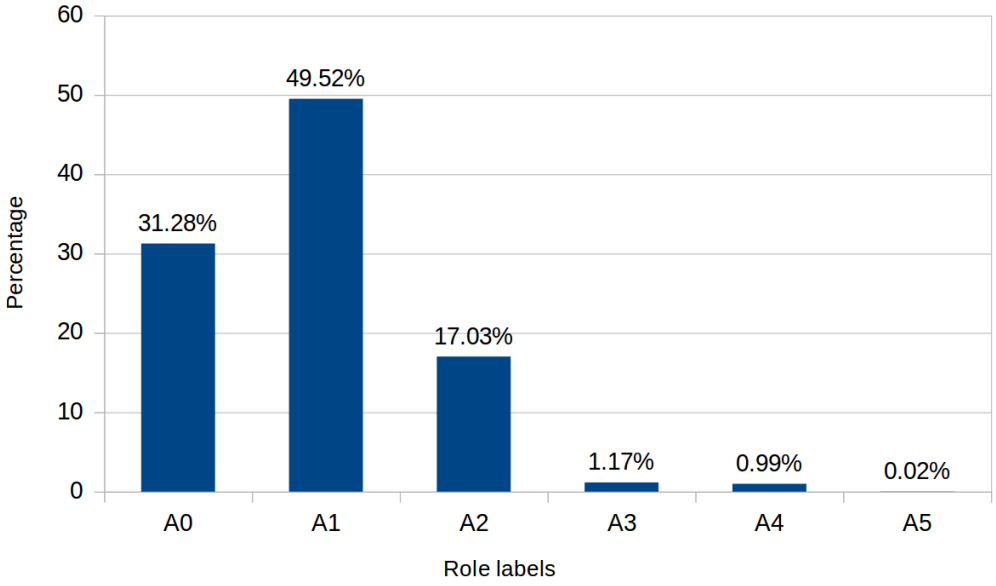
We followed the training-development data split from the shared task. The number of sentences and tokens contained in both is shown in the table 4.1.

In the figure 4.3, the distribution of the role labels is presented for the training set (a) and development set (b). In both of the datasets, there is a noticeable predominance of the labels A1 and A0, which make around 80% of all the labels appearing in the datasets. The most frequent is the label A1, which represents around 50% of all the labels, while A0 is the second most frequent label with 30% (31% in development data). The next, A2 label, is already much less frequent, with around 17% in both datasets. The other core roles appear only on a limited amount of arguments - 1% for A3 and A4 roles, while A5 appears only in very rare cases, 0.02% and 0.03% in the training and development data, respectively.

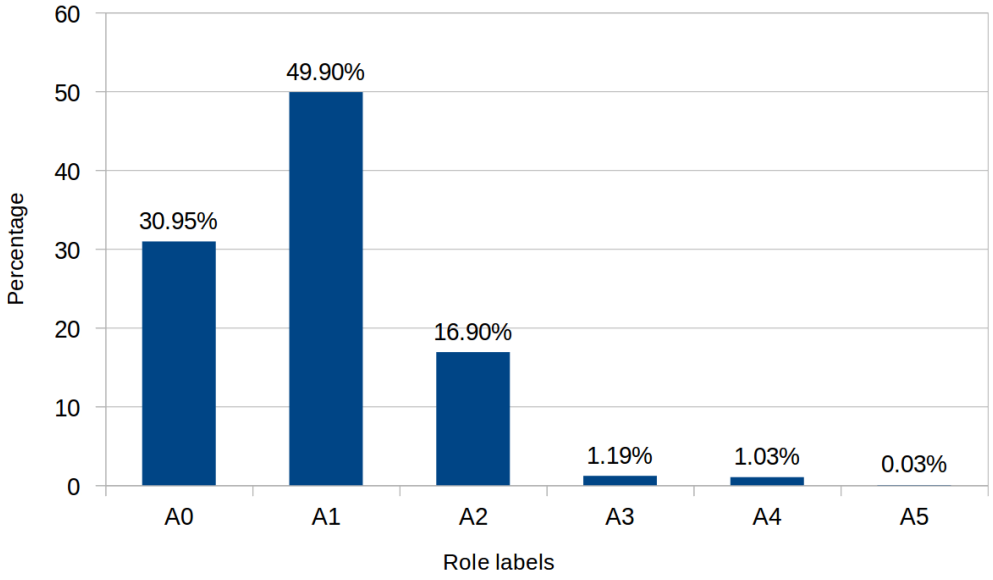
It is clear from this visualization that there will be a bias when it comes to training the models, which will be mostly exposed to the first two most frequent labels. Given that the rest of the labels appear so rarely, the models will be able to encounter them only a limited amount of times and in limited contexts, which can potentially lead to the lack of training of the models for these labels and, thus, lead to them rarely assigning these labels to arguments during the evaluation phase. However, there is a reason for this kind of label distribution: A0 tends to be a PROTO-AGENT, while A1 tends to be a PROTO-PATIENT - two roles that most of the sentences will have. The rest of the roles tend to denote less frequent elements in a sentence - A2 can be a benefactive, instrument, attribute, end point; A3 can be a start point, benefactive, instrument, or attribute; A4 the end point. Besides being more rate, the A2-A5 labels tend to be also inconsistent when it comes to the role they denote, which could also be a potential problem for the classifier.

4.3.2 Test Data

In order to be able to test the models, the data should be annotated using the same set of semantic roles and also following the same annotation guidelines. This



(a) Training data



(b) Development data

Figure 4.3: Distribution of PropBank role labels in the training and development sets.

is usually hard condition to obtain for many languages, however, we follow the previous work (for example, Kozhevnikov and Titov (2013)) which has shown that the languages that fulfill these conditions and that are freely available are Chinese and French, which we will describe in more detail below.

CoNLL-2009 Chinese and French language data		
	Tokens	Sentences
Chinese	2456	71K
French	20K	700

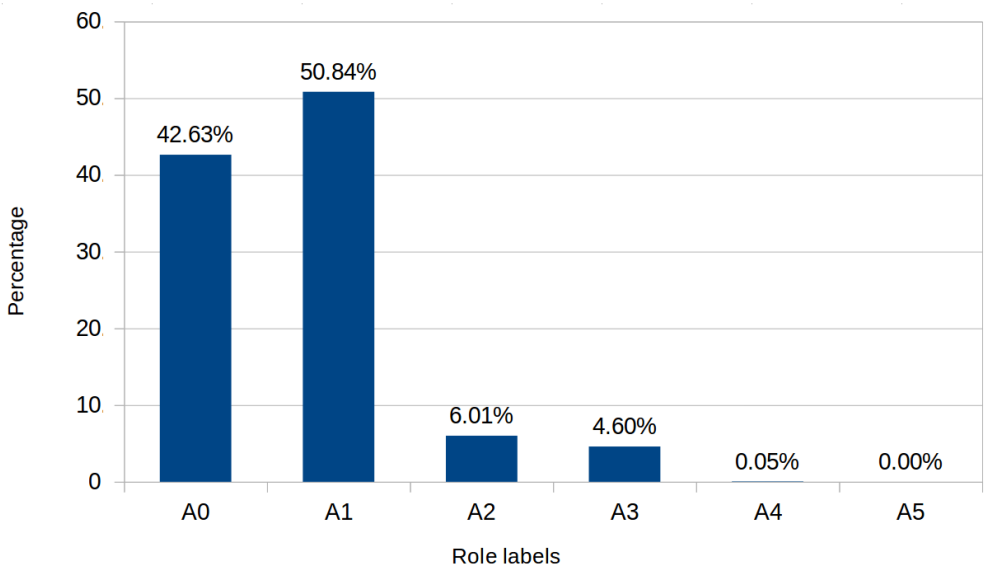
Table 4.2: Number of sentences and tokens in the Chinese and French test data.

The test data for Chinese is the one from the the CoNLL-2009 shared task (Hajič et al., 2009). Even though this annotation of Chinese is not identical to the English one, the guidelines used for the core roles (A0-A5) are similar and interpretable across these two languages (Kingsbury et al., 2004). Since this is the case, we will evaluate our systems on core roles only. On the other hand, for French, we use the data provided by Van der Plas, Merlo, et al. (2011) which contains manually annotated sentences from the Europarl corpus (Koehn, 2005) using PropBank semantic role annotation guidelines. As opposed to the English data, none of the test sets contains nominal predicates, so we exclude these from the evaluation as well. The overview of the number of tokens and sentences in each of these datasets is given in the table 4.2.

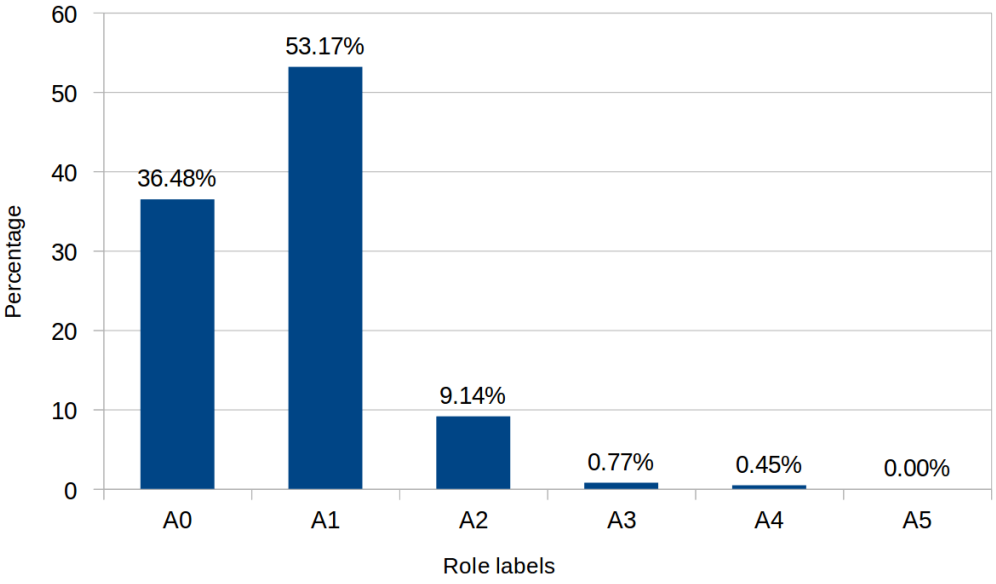
In the figure 4.4 on the next page, the distribution of role labels in the datasets is represented. It is clear that the distribution is similar to the one of the training and development sets. The most frequent labels are A0 and A1, while the rest of the core labels appear much less frequently. A0 and A1 make nearly 43% and 51% in the Chinese data and nearly 37% and 53% in the French data. The main difference here, when compared to the training and the development sets, is that A2 appears around 10% less in the French and Chinese data. Also, in Chinese we can see that A2 and A3 appear 6% and nearly 5% of times, respectively, while the A4 appeared less than 1% and A5 does not appear at all. In the French data, there are almost no A3 and A4 labels appearing (0.77% and 0.45% respectively), while there is no argument containing the A5 role here either. The difference in percentage of the more rare roles (A3-A5) in French as compared to Chinese could be explained by the fact that the French data is significantly smaller - it contains 700 sentences while Chinese test data contains nearly 2500.

In the same way this kind of distribution of labels was problematic for the training set, it could potentially be problematic for testing as well. Given this distribution, we will not have an opportunity to evaluate our models for the more rare labels,

since they do not appear or appear with very low frequency in the test data.



(a) Chinese test data



(b) French test data

Figure 4.4: Distribution of PropBank role labels in the test sets.

4.4 Implementation Setup

We implement our models by using Allennlp library (Gardner et al., 2018) that is freely available[5]. This is a deep learning NLP library built on top of PyTorch. Besides, it made it possible for us to use He, Lee, Lewis, et al. (2017)'s model as they provide the reimplementation of their system using Allennlp library. The original system was implemented in Theano, which we are not familiar with.

	Precision	Recall	F1 score
He et al, 2017 (original)	83.5	83.2	83.4
He et al., 2017 (re-implem.)	78	81	79

Table 4.3: Results of the original system of He, Lee, Lewis, et al. (2017) and its re-implementaiton using Allennlp library.

Given that we are working with a reimplementation of the original system, we ran the original implementation setup in order to get the perception of how differently it performs as compared to the results reported in the paper of He, Lee, Lewis, et al. (2017). The original system uses 100-dimensional pretrained GloVe embeddings, while training and testing on the CoNLL-2012 data following the data split from the CoNLL-2012 Shared task. In the table 4.3, we provide the obtained results. The original paper reports 83.4 F1 score on the CoNLL-2012 test data, while the reimplementation obtains slightly lower F1 score of 79.

Structure of the network. Following the original experiment, our bi-LSTM network contains 8 hidden layers, where four are forward and four reversed LSTMs. There are 300 hidden units, as well as a softmax layer which predicts the distribution of the output.

Initialization. All the weight matrices are initialized with random orthonormal matrices (Saxe et al., 2013), while all tokens are lower-cased and initialized with 300-dimensional cross-lingual FastText embeddings, either by Smith et al. (2017) or Conneau et al. (2018). Those tokens that were not captured by the embeddings are marked with *UNK*, which is randomly initialized.

Training. Following the original experiment, as an optimizer we use Adadelta (Zeiler, 2012) with $p = 0.95$, while the mini-batches size is set to 80. The dropout probability is set to 0.1 and the gradients with the norm larger than 1 are clipped. We set to train all the models in this project for 500 epochs, however there is an early stopping option that is based on the development results (He, Lee, Lewis,

[5]The allennlp library is available here: https://github.com/allenai/allennlp.

et al., 2017). All the experiments are trained on the whole CoNLL-2012 training set and it takes around three days to train on a single GeForce GTX 1050 GPU and around 5 days on a Tesla K20m GPU.

An example of a jsonnet configuration training file is given in the Appendix B.

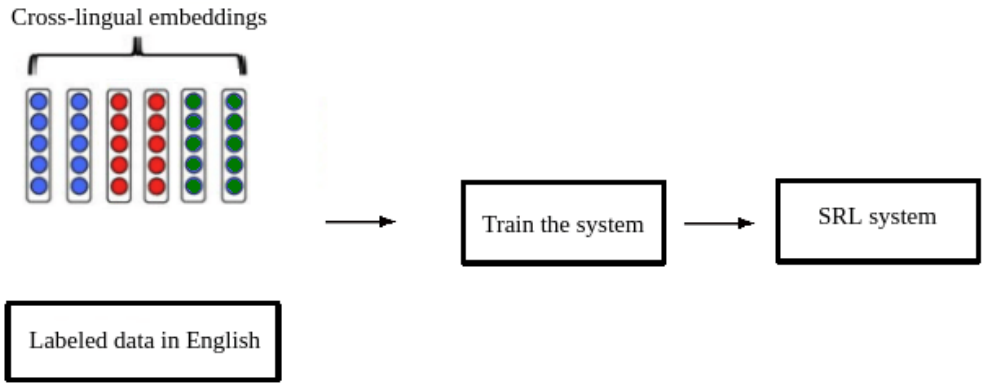
All the models are trained on the English language data using the CoNLL-2012 training and development datasets, while they are evaluated for Chinese and French on the CoNLL-2009 test data and the data from Van der Plas, Merlo, et al. (2011), respectively. This process is illustrated in the figure 4.5 below where we can see that the models are trained using only the English data and the cross-lingual embeddings for the three languages and then tested on the target languages.

4.5 Evaluation

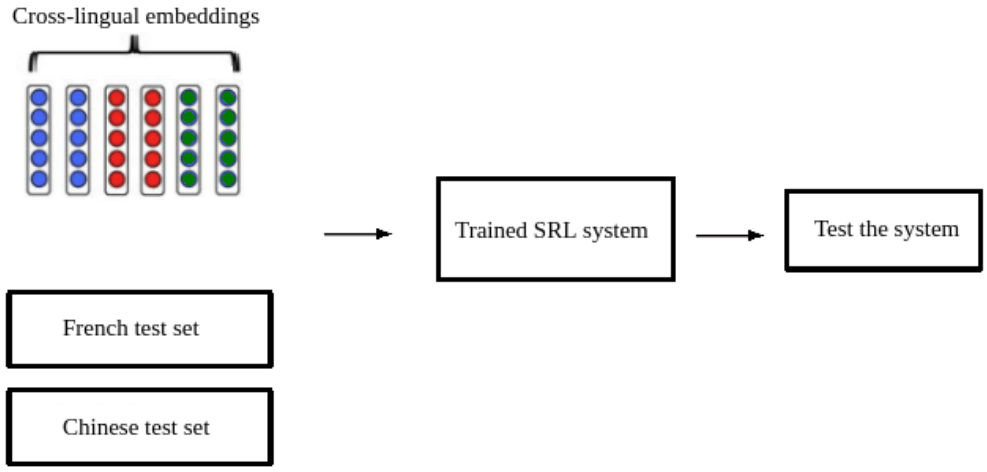
In order to evaluate the classification performance of our systems, the metrics used are precision, recall, macro and weighted F1 measure. We report the results using the weighted F1 as well, because the distribution of the labels in the used datasets is highly skewed, thus, giving more weight to the more represented labels in the datasets (A0, A1, A2). We evaluate the systems only on the core roles (A0-A5), because there is a slight difference in the annotation guidelines for Chinese as compared to the other two languages. Also, given the specificity of our data, which is in fact annotated using two different formalisms, a special type of evaluation methodology is required. In this project, we implement an approach similar to the one of Van der Plas, Merlo, et al. (2011). Our evaluation method consists of three mains steps:

1. Predicate identification - As compared to the previous work, in this project we do not take predicates as given to the system, but rather try to predict the verbal predicates using the POS-taggers for French and Chinese (spaCy (Honnibal and Montani, 2017) and Jieba[6], respectively). As mentioned before, we are not considering nominal predicates here, because they appear only in the English training data, but not in the target languages data and are, thus, excluded. Given that we predict predicates, the first step in the evaluation process is to check how good is the POS-tagger at identifying

[6]Jieba is available here: https://github.com/fxsjy/jieba.



(a) Training on English



(b) Testing on Chinese and French

Figure 4.5: Training and testing process and the data used.

verbs in the sentences for Chinese and French. We consider this to be a very important step, since if the predicate is not identified or not correctly identified, then the classification tends to be missing or incorrect. This is expected given that the semantic roles are defined for each predicate in a sentence. If a predicate is wrongly predicted or missing in the system's prediction output, the error is counted and the wrong predicates are then removed together with their roles from both gold evaluation data and the predicted output. This allows us to focus in the next steps of the evaluation only on those arguments which predicates were well predicted.

2. Argument identification – For the correctly predicted predicates, we evaluate how well did the system do with identifying arguments. This is important because argument identification is a step towards the correct argument

classification. If this was excluded and we were counting only whether the classification is correct or not, we would not know if the system was labeling a random word in a sentence or if it actually learned something about the language structure and identified a true argument and then marked it with a label. In order to achieve this and the next evaluation steps, a certain preprocessing of the data was needed. As described earlier, the data we are using contains two different types of annotation, so we needed to find a way to align these two annotation types. The span-based model would tend to output a label for every element that is included in the constituent, thus, would have a high recall and low precision if evaluated on a dependency-based test set using a standard evaluation approach. What we did to overcome this issue is that we kept only those labels in the predicted output that appear on the head of an argument phrase. This was done by looking at which position in a sentence does the gold data marks the head of a phrase and checks whether this head is also marked in the system's output.

3. Argument classification - This evaluates the classification performance of the system, counting the correctly classified arguments given a predicate as compared to the gold. It is important to note here that we calculate the classification metrics only for those arguments that were correctly identified.

Additional semi-automatic preprocessing steps needed to be made in order to align the two types of formalisms in the datasets. Our evaluation method relies on the exact, line-by-line alignment of the predicted and gold sentences. However, the ConLL-2009 test data had some inconsistencies in the word tokenization. For example, the articles with apostrophes in French are sometimes split (*la protection de l ' environnement*) and sometimes kept together (*la protection de l' environnement*). These and other types of similar errors in both Chinese and French resulted in several thousands mismatched lines that needed to be fixed before conducting the evaluation step.

Chapter 5

Experiments and Results

In this chapter, we will give an overview of the experiments conducted and the results that were obtained. Since the experiments are in accordance with the research questions presented in Chapter 1, we will mention which research question eeach experiment relates to as well.

As it was explained in the previous chapter, we split the evaluation of the systems in three steps: predicate identification, argument identification and argument classification. For each of the steps, we report precision, recall and macro F1, while we also report the weighted F1 score for the classification task. The experiments are grouped in the following way: firstly, we will describe the baseline, language-independent delexicalized system that is trained using the universal Part-of-Speech tags instead of words. Then, we introduce the bilingual systems and their results followed by the description of the multilingual systems. Bilingual systems are trained using aligned bilingual embeddings (for the English-Chinese and English-French language pairs), while multilingual systems make use of aligned multilingual embeddings (English-Chinese-French). Both of these are conducted using aligned FastText embeddings by Smith et al. (2017). Building upon the multilingual experiments, we also make an attempt of training a system for a specific language family, in this case Romance family. The reason why we do not do the same for the Sino-Tibetan family is that neither the pretrained monolingual embeddings were available nor the bilingual dictionaries for training the cross-lingual embeddings. Furthermore, we experimented using MUSE embeddings (Conneau et al., 2018) in all of the previous scenarios in order to investigate how these compare to the ones of Smith et al. (2017) in terms of the systems' performance. And finally, we conclude with a general discussion about the obtained results, which includes a short overview of some of the common errors the systems tend to make, as well as an overview of the difficulties encountered while evaluating the data

and the disadvantages that arised by using this kind of evaluation methodology.

5.1 Baseline - Delexicalized Parser

As the baseline model, we implemented a delexicalized parser, which means that the words in all of the datasets and languages were substituted with their corresponding POS-tags. Even though both training and testing data come with POS-tags already given, these vary slightly across English, Chinese and French languages. For example, in the Chinese dataset, there are POS-tags that do not appear in the others such as M (measure word), VA (predicative adjective), X (numbers and units) (Naiwen Xue et al., 2005); similarly in French, there are CLS (subject clitic pronoun), CLO (object clitic pronoun) or P+D (preposition+determiner), for example, which are specific only to French (Pascal and Sagot, 2012). This was overcome by the use of the set of Universal POS-tags by Petrov et al. (2012) as seen in previous work on dependency parsing (for example McDonald et al. (2011) and projects on grammar induction and projection (Naseem et al., 2010; Zeman and Resnik, 2008)). Thus, we are able to generalize over specificities of POS-tags in different datasets. The universal POS-tags include the following set of tags: NOUN (nouns), VERB (verbs), ADJ (adjectives), ADV (adverbs), PRON (pronouns), DET (determiners), ADP (prepositions or postpositions), NUM (numerals), CONJ (conjunctions), PRT (particles), PUNC (punctuation marks) and X (a catch-all tag). All language specific tags that appear in our data were then replaced with the above tags in order to have them consistent across different languages and datasets.

Even though we will not make use of POS tags in other experiments, this provides a good baseline, since this was the initial approach for a direct model transfer before the appearance of the distributed feature representations. What we are trying to incorporate is another way of overcoming a lexical gap when transferring a system for semantic parsing from one language to another. Besides, in the field of dependency parsing, it was shown that even delexicalized parsers can achieve competitive results, while also being able to capture significant amount of information about the source language structure (McDonald et al., 2011). With this simple method, we are able to tag directly a target language with the system trained only on the source language. Furthermore, this type of system overcomes the lexical gap between languages without any use of parallel data, since the universal POS-tags provide us with a way to transfer, to a certain extent, both structural and lexical knowledge from one language to the other.

5.1.1 Results and Analysis

In the figure 5.1, we show the results obtained by training the modified system of He et al. 2017 on the delexicalized English language data and tested on the delexicalized French and Chinese test sets. Looking first at the predicate identification results, the difference between the success of the French tagger and of the Chinese one can be observed. While for French, the tagger was able to identify the majority of verbs correctly (with F1 score of 81.73%), the Chinese tagger falls behind significantly, with the F1 score lower for 20 points, achieving the result of 59.29% F1. The results of the predicate identification, logically, tend to be the same or very similar in all the experiments as we are testing on the same data in all of them and using the same POS-taggers.

The situation is similar for the argument identification, where the results on the French data are much higher, achieving an F1 score of 66.45%, while on the Chinese data it is 46.72%. The reason for the system to miss so many arguments in both languages could be explained by the fact that both of them differ in grammatical structure from English, which the system was trained on. The reason could be as simple as the word order in the languages.

Finally, looking at the classification performance, we can observe that the system performed similarly on both data sets. Out of the identified arguments, the system achieves macro F1 score of 32.18 on the French and 34.71 on Chinese. Even though the system recognized only a smaller portion of arguments in Chinese, those that were identified were classified with a slightly higher accuracy than in French. These results are obtained by taking into account every label category as equally important, however, as explained in the previous chapter, we also report the weighted F1 score which gives more weight to the more represented labels in the datasets - A0, A1 and A2. The weighted results change significantly for both language, where the system achieves 75.35 weighted F1 score on the French and 78.82 on the Chinese data.

		Macro			Weighted		
		Precision	Recall	F1 score	Precision	Recall	F1 score
French	Predicate identification	79.93	83.62	81.73			
	Argument identification	67.84	65.12	66.45			
	Argument classification	34.09	32.91	32.18	81.39	71.32	75.35
Chinese	Predicate identification	59.77	58.81	59.29			
	Argument identification	48.39	45.17	46.72			
	Argument classification	35.41	34.03	34.71	87.45	73.67	78.82

Figure 5.1: Results of the baseline system for French and Chinese.

5.2 Bilingual and Multilingual Model Transfer

The bilingual and multilingual experiments try to answer the first set of research questions that are at the same time the most general questions and at the core of the idea of this project: *Can we get competitive performance using a deep bi-LSTM network and cross-lingual word embeddings by Smith et al. (2017) while training the models on the English language data and testing on the target languages, French and Chinese?*

Firstly, we will describe the results obtained by training the bilingual systems, which involve training separate models for the English-French and English Chinese language pairs by using aligned bilingual embeddings by Smith et al. (2017). In these alignment pairs, English vector space is used as a common space on to which the other two languages are aligned.

Then, similarly, we will take a look at the results obtained by training one model for all three languages by using the same embeddings, but this time all aligned to the same, English vector space. We refer to this model as a *multilingual* model, since we train a system that can be applied to more than two languages - English, Chinese and French. In this, and all other experiments, we trained on English and then tested on French and Chinese.

The reason we wanted to train the bilingual models besides the multilingual ones is because we wanted to ascertain whether the performance will differ significantly. If not, it might not be worth training two or more separate neural networks for each language pair instead of training only one system which will use all the target language embeddings aligned to the source language space.

5.2.1 Results and Analysis

The results of the bilingual and multilingual experiments are displayed in the tables 5.2 and 5.3, respectively. In terms of predicate and argument identification, in both settings, the situation is similar to the baseline - French POS-tagger performs significantly better than the Chinese one; it achieves around 81% of F1 score in both bilingual and the multilingual experiments, while the Chinese one achieves around 59% of F1 score in both. Likewise, argument identification is more accurate for French, with F1 score of 60.55% in bilingual and 48.51% in the multilingual setting. Chinese argument identification is similar in both settings, slightly under 40% in the multilingual and 40.33% in the bilingual scenario. Interestingly enough, when it comes to argument classification, the systems per-

form better on Chinese in both scenarios, while being significantly better in the multilingual setting with macro F1 score of 37.62%.

		Macro			Weighted		
		Precision	Recall	F1 score	Precision	Recall	F1 score
English - French	Predicate identification	79.93	83.62	81.73			
	Argument identification	67.51	54.9	60.55			
	Argument classification	32.89	30.74	30.41	80.46	70.44	74.44
English - Chinese	Predicate identification	59.77	58.81	59.29			
	Argument identification	49.37	34.09	40.33			
	Argument classification	31.61	43.66	32.35	87.67	73.79	78.94

Figure 5.2: Results of the bilingual systems trained using aligned bilingual embeddings for English-French and English-Chinese language pairs.

		Macro			Weighted		
		Precision	Recall	F1 score	Precision	Recall	F1 score
French	Predicate identification	79.97	83.42	81.66			
	Argument identification	37.35	69.11	48.51			
	Argument classification	27.03	30.47	27.19	76.61	64.21	68.6
Chinese	Predicate identification	59.87	58.81	59.34			
	Argument identification	34.37	43.64	38.45			
	Argument classification	38.59	37.03	37.62	84.73	75.93	79.86

Figure 5.3: Results for French and Chinese for the system trained using aligned English, French and Chinese cross-lingual embeddings.

The results change drastically when using a weighted F1 measure resulting in the best score for French being in the bilingual setting with 74.44% weighted F1, while the system performs better on Chinese when trained in a multilingual setting, with F1 measure of 79.86%. On another note, when looking at the weighted measure, the ratio of precision and recall changes - for Chinese in the bilingual experiment, for French in the multilingual setting. When giving less weight to the less represented label in the datasets, the precision in both cases increases and becomes higher than the recall.

After having looked at the results, we try to answer to the first set of research questions:

- Can a system based on a deep neural network outperform the results obtained with the traditional machine learning approach used by Kozhevnikov and Titov (2013) for the task of SRL model transfer for English, Chinese and French languages?
- Can it also outperform annotation projection model used for comparison in Kozhevnikov and Titov (2013)?

The work and the results of Kozhevnikov and Titov (2013) were presented in Chapter 3 and, as we have seen, by applying the model transfer method in combination with the transferred syntactic information, they achieve 34.5% F1 score for English-Chinese and F1 score of 61.6% for English-French for argument identification task, while in the argument classification they achieve accuracy of 70.1% and 65.1% for English-Chinese and English-French, respectively. We can conclude that our system outperforms theirs in the task of argument identification for Chinese. This likely happened due to the fact that we do not make use of transferred syntax from English and thus avoid some of the errors that kind of information would introduce given the structural differences between these languages. We also outperform their system on the task of argument classification for both Chinese and French when looking at the accuracy scores, which are 75% for the best Chinese and 70.44% for the best French system in this setting.

The second question relates to the projection method of Kozhevnikov and Titov (2013), which we significantly outperform for the task of argument identification - their annotation projection method achieves 13.9% F1 for Chinese and 43.5% for French. The accuracy of their projection model for the argument identification task is 69.2% for Chinese and 66.1% for French, which is slightly lower result than ours in this setting.

Another research question we attempted to answer was: *How does the model perform without using transferred syntactic information as opposed to Kozhevnikov and Titov (2013), whose model transfer system relies on the transferred syntax*? Given that this system leverages only English training data and the cross-lingual embeddings, it has achieved competitive results and in a lot of settings outperforms the method reporeted by Kozhevnikov and Titov (2013).

		Macro			Weighted		
		Precision	Recall	F1 score	Precision	Recall	F1 score
English - French	Predicate identification	83.5	55.3	66.54			
	Argument identification	66.85	64.97	65.89			
	Argument classification	32.17	27.76	29.41	80.11	69.41	73.37
English - Chinese	Predicate identification	59.77	58.81	59.29			
	Argument identification	39.21	48.76	43.47			
	Argument classification	20.42	27.87	15.52	58.78	43.73	48.74

Figure 5.4: Results of the bilingual systems trained using aligned MUSE bilingual embeddings for English-French and English-Chinese language pairs.

5.3 Bilingual and Multilingual Model Transfer using MUSE embeddings

Following the previous experiments and their results, we also wanted to implement bilingual and multilingual systems that use a different method for aligning crosslingual embeddings. We wanted to see whether a different type of embeddings would be more suited for this task and lead to a better performance of the systems in the bilingual and multilingual settings.

For this purpose, we chose the state-of-the art MUSE embeddings by Conneau et al. (2018). The aligned embeddings for English and French were already freely available, while we had to align the ones for Chinese. This was done in a supervised way by following the instructions in the GitHub repository of the MUSE embeddings and by making use of the pretrained monolingual embeddings for English and Chinese, as well as the corresponding bilingual dictionaries.

5.3.1 Results and Analysis

The results of the described experiments are shown in the table 5.4 for bilingual and in the tabe 5.5 for the multilingual models. While the predicate identification remains the same as in the previous experiments, the argument identification improves significantly for French for nearly 5% in both bi- and multilingual experiments as compared to the bilingual English-French with the embeddings of Smith et al. (2017), which was the previously best achieved result. Also, using these embeddings, French results improve in the multilingual setting, from 27.19% to 32.71% of macro F1 score and from 68.6% to 72.47% of weighted F1 score. A very noticeable difference when using these embeddings comes for Chinese, whose performance drastically drops in the bilingual setting, from 32.35% to 15.52% macro

		Macro			Weighted		
		Precision	Recall	F1 score	Precision	Recall	F1 score
French	Predicate identification	83.5	55.3	66.54			
	Argument identification	70.54	60.71	65.26			
	Argument classification	33.56	33.15	32.71	75.63	70.52	72.47
Chinese	Predicate identification	59.77	58.81	59.29			
	Argument identification	46.63	42.25	44.33			
	Argument classification	35.7	35.12	33.86	64.71	67.41	63.91

Figure 5.5: Results for French and Chinese for the system trained using aligned English, French and Chinese MUSE cross-lingual embeddings.

F1 and from 78.94% to 48.74% of weighted F1 score, while also being significantly lower in the multilingual setting, dropping from 37.62% to 33.86% macro F1 and from 79.86% weighted F1 to 63.91%. The reason for this could be the fact that the Chinese crosslingual embeddings were not already available and were, thus, trained by us. Even though we followed all the instructions given in the mentioned repository, the trained embeddings proved to to be of a much lower quality than the pretrained ones for French and English.

This set of experiments also answers the next research question: *Does using monolingual word embeddings for the three languages aligned to a common vector space with different techniques make a significant difference in the obtained results?* As obtained results show, depending on the language and the scenario, they introduce different performance. While they did have a certain improvement for French, they reduced the performance of Chinese.

5.4 Model Transfer within the Romance Language Family

The final set of experiments conducted in order to try to improve the results focused on training a system using aligned embeddings for English and languages of one language family. In this case, the system is trained using languages from a Romance language family aligned to the English vector space. These languages include: French, Spanish, Portuguese and Catalan. The idea behind this is that several languages which have similar lexicons and structure could help one another in the cross-lingual transfer setting. We conduct this experiment only for the Romance language family, because there are no pretrained cross-lingual embeddings from the Sino-tibet family besides Chinese (Mandarin).

We trained our models using aligned embeddings for languages from a Romance language family by both Smith et al. (2017) and Conneau et al. (2018). It is worth to point out that even though we are trying to build an SRL parser only for French, by training a model with cross-lingual embeddings for all of these languages, we are consequently training parsers for other aligned languages as well.

5.4.1 Results and Analysis

The results of the experiments are given in the table 5.6. We can observe that the performance for French decreased when training the system with all of these

languages with both types of cross-lingual embeddings. Even though when looking at the macro averages the performance is comparable for the two types of embeddings (28.39% F1 and 29.65% F1 with Smith et al.'s (2017) and the MUSE, respectively), when looking at the weighted average, the systems' performance differs significantly: 55.04% with Smith et al. (2017) and 67.31% with MUSE embeddings. By looking to the confusion matrices for these two systems we conclude that the performance of MUSE embeddings is justified by the fact that the system trained with these embeddings makes considerably fewer errors for the most common roles (A0, A1, A2).

		Macro			Weighted		
		Precision	Recall	F1 score	Precision	Recall	F1 score
Smith et al. 2017	Predicate identification	79.91	84.47	82.13			
	Argument identification	68.04	56.65	61.83			
	Argument classification	28.69	29.2	28.39	55.27	55.97	55.04
MUSE	Predicate identification	83.5	55.3	66.54			
	Argument identification	48.71	63.65	55.19			
	Argument classification	30.31	33.91	29.65	71.17	64.83	67.31

Figure 5.6: Results for the French language for system trained using aligned English and Romance languages embeddings.

And finally, answering the last research question: *Does incorporating more structurally similar languages to a common vector space result in the boost of the performance?* - the answer is that in our case and using these two types of embeddings, incorporating more similar languages to the systems does not increase the performance of the systems, it in fact reduces the performance.

5.5 Discussion

Summing up all the previously described results, we can conclude that the best system for French was the bilingual one using Smith et al.'s (2017) cross-lingual embeddings and achieving an F1 score of 60.55% on the argument identification and 74.44% of weighted F1 on the argument classification task. On the other hand, the best system for Chinese is the multilingual one using the same type of embeddings which obtained 38.45% F1 for argument identification and 79.86% weighted F1 for the argument classification task. Both of these results are, however, performing similarly to the delexicalized baseline. The best Chinese system outperforms the baseline slightly in terms of argument classification (79.86% versus 78.92% of weighted F1), but it falls behind in terms of argument identification for around 8%. When comparing the best French system to the baseline, we can

notice that the results of the latter are still slightly higher in both of the tasks - approximately 1% for argument classification and 6% for argument identification.

The reason why the bilingual and multilingual systems perform very similarly to the baseline could be explained by the fact that even though the latter does not include any lexical information, it does still incorporate a certain amount of structural information, which is very beneficial for this task. The difference in the outputed results is, thus, not very different. By looking at the outputs of both delexicalized and lexicalized systems, we can notice that they all make similar errors. This could be explained by the fact that both of the system types were trained only on English data and, therefore, learned the English language structure that was then applied to Chinese and French data as well. For example, we can notice that the pattern A0-V-A1 is the most commonly outputted by the system and indeed the most common in the training data. However, this word order is not necessarily correct for Chinese and French. An example of such a case can be seen in the example 7 below. We can see that instead of marking the first mention of *services* with the A1 role and the second with A3, our systems outputs roles A0 and A1, respectively. Even though additional, non-core roles appear in the original gold data, they were removed before testing and are thus not in our output since we are only taking into account the core roles (A0-A5). The third sentence represents the translation of the French examples.

(7)

FR (gold)	Les [$_{A1}$ services] postaux doivent [$_{V}$ rester] des [$_{A3}$ services] publics.
FR (transfer)	Les [$_{A0}$ services] postaux doivent [$_{V}$ rester] des [$_{A1}$ services] publics.
EN (translation)	Postal [$_{A1}$ services] must [$_{V}$ continue] to be [$_{A3}$ public] services.

Also, there is a noticeable difference in the correctness of the outputted labels in shorter and longer sentences. This makes sense, since longer sentences tend to be more complex and require more structural knowledge about a language than the shorter ones. Besides, in longer sentences, a bigger variety of roles appears as opposed to the shorter ones which, in most cases, include A0 and A1 roles only.

Furthermore, the lower macro F1 scores across the systems could be potentially explained by several difficulty factors our systems deal with in terms of the pre-trained embeddings and the type of training and testing data. On the one hand we have the English data used for the training which is comprised of the texts from news, broadcast, talk shows, etc. while the test data comes from the Europarl corpus, which is made out of the proceedings of the European Parliament.

The two types of data come from significantly different domains, which results in different vocabularies and significantly different sentence structures between the two. On top of that, the embeddings that we are using, incorporate the lexicon from Wikipedia, which they were trained on. Thus, we have in fact not only cross-language knowledge transfer, but also cross-domain transfer, which adds an additional layer of difficulty for the systems. And on the other hand, the data used comes in two different annotation formalisms - span and dependency-based - which could not be directly compared without heavily preprocessing the data beforehand.

And finally, as opposed to the previous work on SRL transfer (Kozhevnikov and Titov, 2013) we do a slightly more difficult task as our system does not take predicates as given but tries to predict them by using a POS-tagger. Predicate identification is one of the very important steps in the SRL pipeline, since it is not possible to correctly assign semantic roles to incorrectly identified predicates. It is clear that by attempting to do such a task automatically, we make place for additional errors to be made by the systems as the POS-taggers do not always perform well, especially for Chinese. The reason for the decrease in the taggers performance could be that we applied them to a test set in a different domain (Europarl) than the one it was originally trained on. On top of that, we do not, however, perform a predicate disambiguation task which has proven problematic as different senses of predicates can require a different set of semantic roles.

Chapter 6

Conclusion and Future work

6.1 Conclusion

In this thesis, we constructed several SRL systems for two target languages, French and Chinese, by transferring the state-of-the-art neural model of He, Lee, Lewis, et al. (2017). This was achieved by training the models solely on the English language data and then directly applying them to the target language data. In order to reduce the lexical gap and enable knowledge transfer between these languages, we used pretrained cross-lingual embeddings for English, Chinese and French by Smith et al. (2017) and Conneau et al. (2018). On the CoNLL-2009 data, our best models achieve weighted F1 score of 74.44% and 79.86% for French and Chinese language, respectively. This is a competitive result when compared to the previous work on SRL model transfer of Kozhevnikov and Titov (2013), especially taking into account the fact that we do not use any syntactic information in our experiments.

Detailed results of the main experiments presented in the previous chapter are summarized in table 6.2, while the results of Kozhevnikov and Titov (2013) are given in table 6.1 for easier comparison. Both tables are given at the end of this subsection.

Throughout our project, we attempt to answer four research questions which are in accordance with different approaches we take to train our models, each of which tries to explore the topic of SRL model transfer from a different perspective. In order to summarize our findings, we go again through the research questions while answering each of them with the main conclusions drawn from our experiments.

- *Research question 1:*
  - Can a system based on a deep neural network outperform the results obtained with the traditional machine learning approach used by Kozhevnikov and Titov (2013) for the task of SRL model transfer for the English, Chinese and French language?
  - Given that their model does not always outperform the annotation projection method used as a baseline, the second question is consequently: can our system also outperform the annotation projection model of Kozhevnikov and Titov (2013)?

Our best systems for French and Chinese outperform the ones reported by Kozhevnikov and Titov (2013) in most cases on both argument identification and argument classification tasks. For argument identification, the authors report F1 score of 61.6% for French and 34.5% for Chinese. Our best model for Chinese surpasses this result by around 4% (achieving 38.45% F1 score), while the performance of our best French model for argument identification results in 1% lower performance (60.55% F1 score). On the other hand, when comparing the argument classification results, our best models outperform those of Kozhevnikov and Titov (2013) when taking into account the accuracy scores of the models. Since this is the only measure the authors report for the task of argument classification, it is the only one we are able to compare to. The accuracy scores of our best models for Chinese and French are 75.93% and 70.44%, respectively, while the authors report 70.1% for Chinese and 65.1% for French on the task of model transfer. Consequently, our models preform better than the annotation projection baseline of Kozhevnikov and Titov (2013), which achieves 69.2% accuracy score for Chinese and 66.1% for French. Even though the numbers reported here show higher results for our systems, the accuracy scores could be masking a part of the misperformance of our models shown by the macro F1 scores reported in table 6.2.

- *Research question 2 - Syntactic information:*
  - How does the model perform without any syntactic information as compared to the previous work?

Given the previously mentioned results, we can conclude that the system of He, Lee, Lewis, et al. (2017) used for the model transfer and which does not use any syntactic information achieves very competitive results for this type of task on English, Chinese and French. Given the fact that the related work we described in Chapter 3 has shown that the use of syntactic information tends to significantly boost the performance of SRL systems, in the future it would be interesting to

try to transfer syntactic information as well from English to Chinese and French, and investigate how this change would affect the performance of our systems.

- *Research question 3 - Language similarity:*
  - Does incorporating only structurally similar languages to a common vector space result in a boost in performance?

In order to answer this question, we trained a model using aligned cross-lingual embeddings for English and four languages from the Romance language family. We found that incorporating more languages results in the decrease of the performance for French no matter which type of embeddings are used. This could possibly be explained by the fact that even though these languages are structurally similar, having so many languages aligned to the same vector space introduces more noise to the data and thus results in the decrease of the performance.

- *Research question 4 - Word embeddings alignments:*
  - Does using monolingual word embeddings for the three languages aligned to a common vector space with different techniques make a significant difference in the obtained results?

According to the recent survey of cross-lingual embeddings by Ruder et al. (2017), differently trained embeddings are evaluated for specific tasks and do not guarantee the same results when used in other tasks or with different systems and data. Therefore, we wanted to compare two types of available embeddings, the ones by Smith et al. (2017) on the one hand and the ones by Conneau et al. (2018) on the other. Incorporating MUSE embeddings by Conneau et al. (2018) resulted in the 5% increase in the performance on the task of argument identification for French, as compared to the experiments in which we used the embeddings by Smith et al. (2017). However, the performance for the Chinese language drastically reduces when incorporating MUSE embeddings, which confirms to some extent the findings of Ruder et al. (2017) that not all the pretrained embeddings are equally successful in every task and for every language.

In conclusion, the main contribution of our approach is that it trains and makes available a low-resource SRL parser for French and Chinese by leveraging the existence of large amount of labelled English language data and pretrained cross-lingual word embeddings. As opposed to the previous work, we do not use syntactic information nor parallel corpora. Also, we do not take predicates as given, but incorporate a POS-tagger to identify verbal predicates in target language sentences, which is significantly different and more complex task as compared to approaches applied in previous work.

		Syntax	Transfer	Projection
English-French	Argument identification	Transferred	61.6	43.5
	Argument classification	Transferred	65.1	66.1
English-Chinese	Argument identification	Transferred	34.5	13.9
	Argument classification	Transferred	70.1	69.2

Figure 6.1: SRL model transfer and annotation projection results of Kozhevnikov and Titov (2013) for the tasks of argument identification (F1 score) and argument classification (accuracy).

			Micro			Macro			Weighted		
French systems			Precision	Recall	F1 score	Precision	Recall	F1 score	Precision	Recall	F1 score
Baseline (En-Fr-Zh)		Predicate identification				79.93	83.62	81.73			
		Argument identification				67.84	65.12	66.45			
		Argument classification	71.32	71.32	71.32	34.09	32.91	32.18	81.39	71.32	75.35
Bilingual (En-Fr)	Smith et al. (2017)	Predicate identification				79.93	83.62	81.73			
		Argument identification				67.51	54.9	60.55			
		Argument classification	70.44	70.44	70.44	32.89	30.74	30.41	80.46	70.44	74.44
	MUSE	Predicate identification				83.5	55.3	66.54			
		Argument identification				66.85	64.97	65.89			
		Argument classification	69.41	69.41	69.41	32.17	27.76	29.41	80.11	69.41	73.37
Multilingual (En-Fr-Zh)	Smith et al. (2017)	Predicate identification				79.97	83.42	81.66			
		Argument identification				37.35	69.11	48.51			
		Argument classification	64.21	64.21	64.21	27.03	30.47	27.19	76.61	64.21	68.6
	MUSE	Predicate identification				83.5	55.3	66.54			
		Argument identification				70.54	60.71	65.26			
		Argument classification	70.52	70.52	70.52	33.56	33.15	32.71	75.63	70.52	72.47
Rromance lang.fam.	Smith et al. (2017)	Predicate identification				79.91	84.47	82.13			
		Argument identification				68.04	56.65	61.83			
		Argument classification	55.97	55.97	55.97	28.69	29.2	28.39	55.27	55.97	55.04
	MUSE	Predicate identification				83.5	55.3	66.54			
		Argument identification				48.71	63.65	55.19			
		Argument classification	64.83	64.83	64.83	30.31	33.91	29.65	71.17	64.83	67.31

(a) Results of all systems tested on French language data.

			Micro			Macro			Weighted		
Chinese systems			Precision	Recall	F1 score	Precision	Recall	F1 score	Precision	Recall	F1 score
Baseline (En-Fr-Zh)		Predicate identification				59.77	58.81	59.29			
		Argument identification				48.39	45.17	46.72			
		Argument classification	73.67	73.67	73.67	35.41	34.03	34.71	87.45	73.67	78.82
Bilingual (En-Zh)	Smith et al. (2017)	Predicate identification				59.77	58.81	59.29			
		Argument identification				49.37	34.09	40.33			
		Argument classification	73.79	73.79	73.79	31.61	43.66	32.35	87.67	73.79	78.94
	MUSE	Predicate identification				59.77	58.81	59.29			
		Argument identification				39.21	48.76	43.47			
		Argument classification	43.73	43.73	43.73	20.42	27.87	15.52	58.78	43.73	48.74
Multilingual (En-Fr-Zh)	Smith et al. (2017)	Predicate identification				59.87	58.81	59.34			
		Argument identification				34.37	43.64	38.45			
		Argument classification	75.93	75.93	75.93	38.59	37.03	37.62	84.73	75.93	79.86
	MUSE	Predicate identification				59.77	58.81	59.29			
		Argument identification				46.63	42.25	44.33			
		Argument classification	67.41	67.41	67.41	35.7	35.12	33.86	64.71	67.41	63.91

(b) Results of all systems tested on Chinese language data.

Figure 6.2: Results of all our systems trained on English and tested on French and Chinese language data.

6.2 Future Work

There various directions in which the work conducted in this thesis could be extended in order to improve the obtained results. Most of the following ideas are derived from related literature:

- Fine-tuning the models on the labeled target language data. The previous research has shown that incorporating even a 100 sentences of labelled target data can make a significant improvement to the results of a transfer system (J. Guo et al., 2016).
- Training the cross-lingual embeddings specifically for the task of SRL, or at least train the embeddings on a similar domain to the one of the data sets. As we already mentioned in the theoretical background, Ruder et al. (2017) has recently pointed out that the use of availabe pretrained embeddings does not neccessarily result in equally good performance in every NLP task. The authors showed that training embeddings for a specific task might be a better approach.
- Incorporating a better method of predicting predicates. Using a POS-tagger for predicate identification introduced a certain amount of errors to our systems. Also, in this way we are not able to recognize nominal predicates, but only verbal ones.
- Related to the previous point, we would like to perform predicate disambiguation before assigning roles to arguments, since different predicate senses can evoke different set of semantic roles.
- If possible, implement the systems using less noisy test data which uses the same annotation format. In our case, we show that a lot of errors were introduced because of the nature of the data we were dealing with, so the cleaner datasets could potentially lead to better results of the same systems.

Bibliography

Abend, O. & Rappoport, A. (2017). The State of the Art in Semantic Representation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (volume 1: Long papers)* (pp. 77–89). Vancouver, Canada: Association for Computational Linguistics.

Akbik, A., Blythe, D., & Vollgraf, R. (2018). Contextual String Embeddings for Sequence Labeling. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 1638–1649). Santa Fe, New Mexico, USA: Association for Computational Linguistics.

Akbik, A., Chiticariu, L., Danilevsky, M., Li, Y., Vaithyanathan, S., & Zhu, H. (2015). Generating High Quality Proposition Banks for Multilingual Semantic Role Labeling. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, *1*, 397–418.

Akbik, A., Guan, X., & Li, Y. (2016). Multilingual Aliasing for Auto-Generating Proposition Banks. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical papers* (pp. 3466–3474). Osaka, Japan: The COLING 2016 Organizing Committee.

Annesi, P. & Basili, R. (2010). Cross-Lingual Alignment of FrameNet Annotations Through Hidden Markov Models. In *Proceedings of the 11th International Conference on Computational Linguistics and Intelligent Text Processing* (pp. 12–25). CICLing'10. Romania: Springer-Verlag.

Aufrant, L., Wisniewski, G., & Yvon, F. (2016). Cross-lingual Alignment Transfer: A Chicken-and-egg Story? In *Proceedings of the Workshop on Multilingual and Cross-lingual Methods in NLP* (pp. 35–44). San Diego, CA, United States.

Baroni, M., Dinu, G., & Kruszewski, G. (2014). Don't Count, Predict! A Systematic Comparison of Context-counting vs. Context-predicting Semantic Vectors. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (volume 1: long papers)* (pp. 238–247). Baltimore, Maryland: Association for Computational Linguistics.

Basili, R., Cao, D. D., Croce, D., Coppola, B., & Moschitti, A. (2009). Cross-Language Frame Semantics Transfer in Bilingual Corpora. *Lecture notes in Computer Science (including subseries lecture notes in Artificial Intelligence and lecture notes in Bioinformatics)*, *5449 LNCS*, 332–345.

Björkelund, A., Hafdell, L., & Nugues, P. (2009). Multilingual Semantic Role Labeling. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task* (pp. 43–48). Boulder, Colorado: Association for Computational Linguistics.

Bohnet, B., McDonald, R., Simões, G., Andor, D., Pitler, E., & Maynez, J. (2018). Morphosyntactic Tagging with a Meta-BiLSTM Model over Context Sensitive Token Encodings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (volume 1: long papers)* (pp. 2642–2652). Melbourne, Australia: Association for Computational Linguistics.

Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, *5*, 135–146.

Cai, J., Jiang, Y., & Tu, K. (2017). CRF Autoencoder for Unsupervised Dependency Parsing. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 1638–1643). Copenhagen, Denmark: Association for Computational Linguistics.

Choi, J. & Palmer, M. (2011). Transition-based Semantic Role Labeling Using Predicate Argument Clustering. Workshop on Relational Models of Semantics.

Clark, K., Luong, T., Manning, C. D., & Le, Q. V. (2018). Semi-Supervised Sequence Modeling with Cross-View Training. Retrieved from https://drive.google.com/file/d/1aOLJRYmhl0ZpeGtYE8jpFE906uETzrM4/view

Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., & Kuksa, P. (2011). Natural Language Processing (Almost) from Scratch. *Journal of Machine Learning Research*, *12*, 2493–2537.

Conneau, A., Lample, G., Ranzato, M., Denoyer, L., & Jégou, H. (2018). Word Translation Without Parallel Data. *Computing Research Repository, abs/1710.04087*.

Deschacht, K. & Moens, M.-F. (2009). Semi-supervised Semantic Role Labeling Using the Latent Words Language Model. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing* (pp. 21–29). Singapore: Association for Computational Linguistics.

Dozat, T. & Manning, C. D. (2016). Deep Biaffine Attention for Neural Dependency Parsing. *Computing Research Repository, abs/1611.01734*.

Duong, L., Cohn, T., Bird, S., & Cook, P. (2015a). Cross-lingual Transfer for Unsupervised Dependency Parsing Without Parallel Data. In *Proceedings of the Nineteenth Conference on Computational Natural Language Learning* (pp. 113–122). Beijing, China: Association for Computational Linguistics.

Duong, L., Cohn, T., Bird, S., & Cook, P. (2015b). Low Resource Dependency Parsing: Cross-lingual Parameter Sharing in a Neural Network Parser. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (volume 2: short papers)* (pp. 845–850). Beijing, China: Association for Computational Linguistics.

Edunov, S., Ott, M., Auli, M., & Grangier, D. (2018). Understanding Back-Translation at Scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (pp. 489–500). Brussels, Belgium: Association for Computational Linguistics.

Fader, A., Soderland, S., & Etzioni, O. (2011). Identifying Relations for Open Information Extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing* (pp. 1535–1545). Edinburgh, Scotland, UK: Association for Computational Linguistics.

Fillmore, C. & Baker, C. (2009). *A Frames Approach to Semantic Analysis*. Oxford University Press.

Firth, J. (1957). A Synopsis of Linguistic Theory 1930-1955. In *Studies in Linguistic Analysis*. Reprinted in Palmer, F. (ed. 1968) Selected Papers of J. R. Firth, Longman, Harlow. Philological Society, Oxford.

FitzGerald, N., Täckström, O., Ganchev, K., & Das, D. (2015). Semantic Role Labeling with Neural Network Factors. In *Proceedings of the 2015 Confer-*

ence on Empirical Methods in Natural Language Processing (pp. 960–970). Lisbon, Portugal: Association for Computational Linguistics.

Fürstenau, H. & Lapata, M. (2012). Semi-Supervised Semantic Role Labeling via Structural Alignment. *Computational Linguistics*, *38*(1).

Gal, Y. & Ghahramani, Z. (2016). A Theoretically Grounded Application of Dropout in Recurrent Neural Networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems* (pp. 1027–1035). NIPS'16. Barcelona, Spain: Curran Associates Inc.

Gao, Q. & Vogel, S. (2011). Corpus Expansion for Statistical Machine Translation with Semantic Role Label Substitution Rules. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies* (pp. 294–298). Portland, Oregon, USA: Association for Computational Linguistics.

Gardner, M., Grus, J., Neumann, M., Tafjord, O., Dasigi, P., F. Liu, N., ... Zettlemoyer, L. (2018). AllenNLP: A Deep Semantic Natural Language Processing Platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)* (pp. 1–6). Melbourne, Australia: Association for Computational Linguistics.

Gildea, D. & Jurafsky, D. (2002). Automatic Labeling of Semantic Roles. *Computational Linguistics*, *28*(3), 245–288.

Gildea, D. & Palmer, M. (2002). The Necessity of Parsing for Predicate Argument Recognition. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics* (pp. 239–246). Philadelphia, Pennsylvania: Association for Computational Linguistics.

Goldberg, Y. (2016). A Primer on Neural Network Models for Natural Language Processing. *Journal of Artificial Intelligence Research*, *57*(1), 345–420.

Graves, A., Mohamed, A., & Hinton, G. E. (2013). Speech Recognition with Deep Recurrent Neural Networks. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 6645–6649.

Grenager, T. & Manning, C. D. (2006). Unsupervised Discovery of a Statistical Verb Lexicon. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing* (pp. 1–8). Sydney, Australia: Association for Computational Linguistics.

Guo, J., Che, W., Yarowsky, D., Wang, H., & Liu, T. (2016). A Representation Learning Framework for Multi-source Transfer Parsing. In *Proceedings of*

the Thirtieth AAAI Conference on Artificial Intelligence (pp. 2734–2740). AAAI'16. Phoenix, Arizona: AAAI Press.

Hajič, J., Ciaramita, M., Johansson, R., Kawahara, D., Martí, M. A., Màrquez, L., ... Zhang, Y. (2009). The CoNLL-2009 Shared Task: Syntactic and Semantic Dependencies in Multiple Languages. *Computational Linguistics*, (June), 1–18.

Harris, Z. (1954). Distributional Structure. *Word*, *10*(23), 146–162.

He, L., Lee, K., Levy, O., & Zettlemoyer, L. (2018). Jointly Predicting Predicates and Arguments in Neural Semantic Role Labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: short papers)* (pp. 364–369). Melbourne, Australia: Association for Computational Linguistics.

He, L., Lee, K., Lewis, M., & Zettlemoyer, L. (2017). Deep Semantic Role Labeling: What Works and What's Next. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: long papers)*, 473–483.

Hermans, M. & Schrauwen, B. (2013). Training and Analyzing Deep Recurrent Neural Networks. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1* (pp. 190–198). Lake Tahoe, Nevada: Curran Associates Inc.

Hochreiter, S. & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780.

Honnibal, M. & Montani, I. (2017). spaCy 2: Natural Language Understanding with Bloom Embeddings, Convolutional Neural Networks and Incremental Parsing. *To appear*.

Jacobson, L. (2013). Introduction to Artificial Neural Networks - Part 1. Retrieved from http://www.theprojectspot.com/tutorial-post/introduction-to-artificial-neural-networks-part-1/7. Blog.

Jiang, Y., Han, W., & Tu, K. (2016). Unsupervised Neural Dependency Parsing. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 763–771). Austin, Texas: Association for Computational Linguistics.

Jurafsky, D. & Martin, J. H. (2018). *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and*

Speech Recognition (3rd (draft)). Retrieved from https://web.stanford.edu/~jurafsky/slp3/

Kaisser, M. & Webber, B. (2007). Question Answering Based on Semantic Roles. In *ACL 2007 Workshop on Deep Linguistic Processing* (pp. 41–48). Prague, Czech Republic: Association for Computational Linguistics.

Khan, A., Salim, N., & Kumar, Y. J. (2015). A Framework for Multi-document Abstractive Summarization Based on Semantic Role Labelling. *Applied Soft Computing*, *30*, 737–747.

Kingsbury, P., Xue, N., & Palmer, M. (2004). Propbanking in Parallel. In *Workshop on the Amazing Utility of Parallel and Comparable Corpora, in conjunction with LREC'04*.

Klementiev, A., Titov, I., & Bhattarai, B. (2012). Inducing Cross-lingual Distributed Representations of Words. In *Proceedings of COLING 2012* (pp. 1459–1474). Mumbai, India: The COLING 2012 Organizing Committee.

Koehn, P. (2005). Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the Tenth Machine Translation Summit* (pp. 79–86). Phuket, Thailand: AAMT.

Kozhevnikov, M. (2017). *Cross-lingual Transfer of Semantic Role Labeling Models* (Doctoral dissertation, Saarland University, Saarbrücken, Germany).

Kozhevnikov, M. & Titov, I. (2013). Cross-lingual Transfer of Semantic Role Labeling Models. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*.

Krause, B., Kahembwe, E., Murray, I., & Renals, S. (2018). Dynamic Evaluation of Neural Sequence Models. In J. Dy & A. Krause (Eds.), *Proceedings of the 35th International Conference on Machine Learning* (Vol. 80, pp. 2766–2775). PMLR.

Lang, J. & Lapata, M. (2010). Unsupervised Induction of Semantic Roles. In *Human language technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics* (pp. 939–947). Los Angeles, California: Association for Computational Linguistics.

Lazaridou, A., Dinu, G., & Baroni, M. (2015). Hubness and Pollution: Delving into Cross-Space Mapping for Zero-Shot Learning. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (volume 1:*

Long papers) (pp. 270–280). Beijing, China: Association for Computational Linguistics.

Lee, K., He, L., Lewis, M., & Zettlemoyer, L. (2017). End-to-end Neural Coreference Resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 188–197). Copenhagen, Denmark: Association for Computational Linguistics.

Lee, K., Lewis, M., & Zettlemoyer, L. (2016). Global Neural CCG Parsing with Optimality Guarantees. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing* (pp. 2366–2376). Austin, Texas: Association for Computational Linguistics.

Lewis, M. & Steedman, M. (2014). A* CCG Parsing with a Supertag-factored Model. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 990–1000). Doha, Qatar: Association for Computational Linguistics.

Ling, W., Dyer, C., Black, A. W., Trancoso, I., Fermandez, R., Amir, S., ... Luis, T. (2015). Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing* (pp. 1520–1530). Lisbon, Portugal: Association for Computational Linguistics.

Liu, D. & Gildea, D. (2010). Semantic Role Features for Machine Translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)* (pp. 716–724). Beijing, China: COLING 2010 Organizing Committee.

Lo, C., Addanki, K., Saers, M., & Wu, D. (2013). Improving Machine Translation by Training Against an Automatic Semantic Frame Based Evaluation Metric. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (volume 2: Short papers)* (pp. 375–381). Sofia, Bulgaria: Association for Computational Linguistics.

Luan, Y., Ji, Y., Hajishirzi, H., & Li, B. (2016). Multiplicative Representations for Unsupervised Semantic Role Induction. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (volume 2: Short papers)* (pp. 118–123). Berlin, Germany: Association for Computational Linguistics.

Maqsud, U., Arnold, S., Hülfenhaus, M., & Akbik, A. (2014). Nerdle: Topic-Specific Question Answering Using Wikia Seeds. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Sys-*

tem demonstrations (pp. 81–85). Dublin, Ireland: Dublin City University and Association for Computational Linguistics.

Marasović, A. & Frank, A. (2018). SRL4ORL: Improving Opinion Role Labeling Using Multi-Task Learning with Semantic Role Labeling. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, volume 1 (long papers)* (pp. 583–594). New Orleans, Louisiana: Association for Computational Linguistics.

Marcheggiani, D., Frolov, A., & Titov, I. (2017). A Simple and Accurate Syntax-Agnostic Neural Model for Dependency-based Semantic Role Labeling. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)* (pp. 411–420). Vancouver, Canada: Association for Computational Linguistics.

Marcheggiani, D., Roth, M., Titov, I., & Van Durme, B. (2017). Semantic Role Labeling Tutorial Part 2: Neural Methods for Semantic Role Labeling. EMNLP 2017. Copenhagen.

Marcheggiani, D. & Titov, I. (2017). Encoding Sentences with Graph Convolutional Networks for Semantic Role Labeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing* (pp. 1506–1515). Copenhagen, Denmark: Association for Computational Linguistics.

Màrquez, L., Carreras, X., Litkowski, K. C., & Stevenson, S. (2008). Semantic Role Labeling: An Introduction to the Special Issue. *Computational Linguistics*, *34*(2), 145–159.

McDonald, R., Petrov, S., & Hall, K. (2011). Multi-source Transfer of Delexicalized Dependency Parsers. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 62–72). Edinburgh, United Kingdom: Association for Computational Linguistics.

Merlo, P. & Van Der Plas, L. (2009). Abstraction and Generalisation in Semantic Role Labels: PropBank, VerbNet or Both? 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: Volume 1 - volume 1* (pp. 288–296). ACL '09. Suntec, Singapore: Association for Computational Linguistics.

Meyers, A., Reeves, R., Macleod, C., Szekely, R., Zielinska, V., Young, B., & Grishman, R. (2004). Annotating Noun Argument Structure for NomBank. In *Proceedings of the Fourth International Conference on Language Re-*

sources and Evaluation (LREC'04). Lisbon, Portugal: European Language Resources Association (ELRA).

Mikolov, T., Chen, K., Corrado, G. S., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *CoRR*, *abs/1301.3781*.

Mikolov, T., Le, Q. V., & Sutskever, I. (2013). Exploiting Similarities among Languages for Machine Translation. *CoRR*, *abs/1309.4168*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G., & Dean, J. (2013). Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2* (pp. 3111–3119). Lake Tahoe, Nevada: Curran Associates Inc.

Naseem, T., Chen, H., Barzilay, R., & Johnson, M. (2010). Using Universal Linguistic Knowledge to Guide Grammar Induction. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing* (pp. 1234–1244). Cambridge, MA: Association for Computational Linguistics.

Olah, C. (2015). Understanding LSTM Networks. Retrieved from http://colah.github.io/posts/2015-08-Understanding-LSTMs/. Blog.

Padó, S. (2007). *Cross-lingual Annotation Projection Models for Role-semantic Information* (Doctoral dissertation, Saarbrucken, Germany).

Padó, S. & Lapata, M. (2009). Cross-lingual Annotation Projection of Semantic Roles. *The Journal of Artificial Intelligence Research*, *36*, 307–340.

Palmer, M., Gildea, D., & Kingsbury, P. (2005). The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, *31*(1), 71–106.

Pascal, D. & Sagot, B. (2012). Coupling an Annotated Corpus and a Lexicon for State-of-the-art POS Tagging. *Language Resources and Evaluation*, *46*(4), 721–736.

Pascanu, R., Gülçehre, Ç., Cho, K., & Bengio, Y. (2014). How to Construct Deep Recurrent Neural Networks. *CoRR*, *abs/1312.6026*.

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532–1543). Doha, Qatar: Association for Computational Linguistics.

Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, volume 1 (long papers)* (pp. 2227–2237). New Orleans, Louisiana: Association for Computational Linguistics.

Petrov, S., Das, D., & McDonald, R. (2012). A Universal Part-of-Speech Tagset. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*. Istanbul, Turkey: European Language Resources Association (ELRA).

Pradhan, S., Moschitti, A., Xue, N., Uryupina, O., & Zhang, Y. (2012). CoNLL-2012 Shared Task: Modeling Multilingual Unrestricted Coreference in OntoNotes. In *Joint Conference on EMNLP and CoNLL - Shared Task* (pp. 1–40). Jeju, Republic of Korea: Association for Computational Linguistics.

Pradhan, S., Ward, W., Hacioglu, K., Martin, J. H., & Jurafsky, D. (2005). Semantic Role Labeling Using Different Syntactic Views. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics* (pp. 581–588). Ann Arbor, Michigan: Association for Computational Linguistics.

Roth, M. & Lapata, M. (2016). Neural Semantic Role Labeling with Dependency Path Embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1192–1202). Berlin, Germany: Association for Computational Linguistics.

Ruder, S., Vulić, I., & Søgaard, A. (2017). A Survey Of Cross-lingual Word Embedding Models. *The Journal of Artificial Intelligence Research*.

Saxe, A. M., McClelland, J. L., & Ganguli, S. (2013). Exact Solutions to the Nonlinear Dynamics of Learning in Deep Linear Neural Networks. *CoRR*, *abs/1312.6120*.

Schönemann, P. H. (1966). A Generalized Solution of the Orthogonal Procrustes Problem. *Psychometrika*, *31*(1), 1–10.

Schuler, K. K. (2005). *VerbNet: A Broad-coverage, Comprehensive Verb Lexicon* (Doctoral dissertation, Philadelphia, PA, USA).

Schuster, M. & Paliwal, K. K. (1997). Bidirectional Recurrent Neural Networks. *IEEE Transactions on Signal Processing*, *45*, 2673–2681.

Shen, D. & Lapata, M. (2007). Using Semantic Roles to Improve Question Answering. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*.

Smith, S. L., Turban, D. H. P., Hamblin, S., & Hammerla, N. Y. (2017). Offline Bilingual Word Vectors, Orthogonal Transformations and the Inverted Softmax. *CoRR, abs/1702.03859*.

Srivastava, R. K., Greff, K., & Schmidhuber, J. (2015). Training Very Deep Networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2* (pp. 2377–2385). Montreal, Canada: MIT Press.

Surdeanu, M., Harabagiu, S., Williams, J., & Aarseth, P. (2003). Using Predicate-argument Structures for Information Extraction. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1* (pp. 8–15). Sapporo, Japan: Association for Computational Linguistics.

Swier, R. S. & Stevenson, S. (2004). Unsupervised Semantic Role Labelling. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*.

Täckström, O., McDonald, R., & Uszkoreit, J. (2012). Cross-lingual Word Clusters for Direct Transfer of Linguistic Structure. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 477–487). Montrèal, Canada: Association for Computational Linguistics.

Tan, Z., Wang, M., Xie, J., Chen, Y., & Shi, X. (2018). Deep Semantic Role Labeling With Self-Attention. In *AAAI*.

Titov, I. & Khoddam, E. (2015). Unsupervised Induction of Semantic Roles within a Reconstruction-Error Minimization Framework. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 1–10). Denver, Colorado: Association for Computational Linguistics.

Titov, I. & Klementiev, A. (2012a). A Bayesian Approach to Unsupervised Semantic Role Induction. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics* (pp. 12–22). Avignon, France: Association for Computational Linguistics.

Titov, I. & Klementiev, A. (2012b). Crosslingual Induction of Semantic Roles. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (volume 1: Long papers)* (pp. 647–656). Jeju Island, Korea: Association for Computational Linguistics.

Titov, I. & Klementiev, A. (2012c). Semi-Supervised Semantic Role Labeling: Approaching from an Unsupervised Perspective. In *Proceedings of COLING 2012* (pp. 2635–2652). Mumbai, India: The COLING 2012 Organizing Committee.

Trandabât, D. (2011). Using Semantic Roles to Improve Summaries. In *Proceedings of the 13th European Workshop on Natural Language Generation* (pp. 164–169). Nancy, France: Association for Computational Linguistics.

Upadhyay, S., Faruqui, M., Dyer, C., & Roth, D. (2016). Cross-lingual Models of Word Embeddings: An Empirical Comparison. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 1661–1670). Berlin, Germany: Association for Computational Linguistics.

Van der Plas, L., Apidianaki, M., & Chen, C. (2014). Global Methods for Cross-lingual Semantic Role and Predicate Labelling. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers* (pp. 1279–1290). Dublin, Ireland: Dublin City University and Association for Computational Linguistics.

Van der Plas, L., Henderson, J., & Merlo, P. (2009). Domain Adaptation with Artificial Data for Semantic Parsing of Speech. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, companion volume: Short papers* (pp. 125–128). Boulder, Colorado: Association for Computational Linguistics.

Van der Plas, L., Merlo, P., & Henderson, J. (2011). Scaling Up Automatic Cross-lingual Semantic Role Annotation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2* (pp. 299–304). HLT '11. Portland, Oregon: Association for Computational Linguistics.

Van der Plas, L., Samardzic, T., & Merlo, P. (2010). Cross-Lingual Validity of PropBank in the Manual Annotation of French. In *Proceedings of the Fourth Linguistic Annotation Workshop* (pp. 113–117). Uppsala, Sweden: Association for Computational Linguistics.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... Polosukhin, I. (2017). Attention is All you Need. In *Advances in Neural Information Processing Systems 30* (pp. 5998–6008). Curran Associates, Inc.

Vulić, I. (2017). Cross-Lingual Syntactically Informed Distributed Word Representations. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers* (pp. 408–414). Valencia, Spain: Association for Computational Linguistics.

Xiao, M. & Guo, Y. (2014). Distributed Word Representation Learning for Cross-Lingual Dependency Parsing. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning* (pp. 119–129). Ann Arbor, Michigan: Association for Computational Linguistics.

Xue, N. [Naiwen], Xia, F., Chiou, F., & Palmer, M. (2005). The Penn Chinese TreeBank: Phrase Structure Annotation of a Large Corpus. *Natural Language Engineering*, *11*(2), 207–238.

Xue, N. [Nianwen] & Palmer, M. (2004). Calibrating Features for Semantic Role Labeling. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*.

Yang, Z., Dai, Z., Salakhutdinov, R., & Cohen, W. W. (2017). Breaking the Softmax Bottleneck: A High-Rank RNN Language Model. *CoRR*, *abs/1711.03953*.

Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent Trends in Deep Learning Based Natural Language Processing [Review Article]. *IEEE Computational Intelligence Magazine*, *13*, 55–75.

Zadeh, S., Kaljahi, R., & Baba, M. S. (2011). Investigation of Co-training Views and Variations for Semantic Role Labeling. In *Proceedings of Workshop on Robust Unsupervised and Semi-supervised Methods in Natural Language Processing* (pp. 41–49). Hissar, Bulgaria: Association for Computational Linguistics.

Zapirain, B., Agirre, E., & Màrquez, L. (2007). UBC-UPC: Sequential SRL Using Selectional Preferences: An Aproach with Maximum Entropy Markov Models. In *Proceedings of the 4th International Workshop on Semantic Evaluations* (pp. 354–357). Prague, Czech Republic: Association for Computational Linguistics.

Zeiler, M. D. (2012). ADADELTA: An Adaptive Learning Rate Method. *CoRR*, *abs/1212.5701*.

Zeman, D. & Resnik, P. (2008). Cross-Language Parser Adaptation between Related Languages. In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*.

Zhou, J. & Xu, W. (2015). End-to-end Learning of Semantic Role Labeling Using Recurrent Neural Networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)* (pp. 1127–1137). Beijing, China: Association for Computational Linguistics.

Zilly, J. G., Srivastava, R. K., Koutnık, J., & Schmidhuber, J. (2017). Recurrent Highway Networks. In *Proceedings of the 34th International Conference on Machine Learning* (Vol. 70, pp. 4189–4198). International Convention Centre, Sydney, Australia: PMLR.

Zou, W. Y., Socher, R., Cer, D., & Manning, C. D. (2013). Bilingual Word Embeddings for Phrase-Based Machine Translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing* (pp. 1393–1398). Seattle, Washington, USA: Association for Computational Linguistics.

Appendix A

Chapter 4.3

Column	Type	Description
1	Document ID	This is a variation on the document filename
2	Part number	Some files are divided into multiple parts numbered as 000, 001, 002, ... etc.
3	Word number	This is the word index in the sentence
4	Word	The word itself
5	Part of Speech	Part of Speech of the word
6	Parse bit	This is the bracketed structure broken before the first open parenthesis in the parse, and the word/part-of-speech leaf replaced with a *. The full parse can be created by substituting the asterisk with the `([pos] [word])` string (or leaf) and concatenating the items in the rows of that column.
7	Lemma	The predicate/sense lemma is mentioned for the rows for which we have semantic role or word sense information. All other rows are marked with a –
8	Predicate Frameset ID	This is the PropBank frameset ID of the predicate in Column 7.
9	Word sense	This is the word sense of the word in Column 4.
10	Speaker/Author	This is the speaker or author name where available. Mostly in Broadcast Conversation and Weblog data.
11	Named Entities	These columns identifies the spans representing various named entities.
12:N	Predicate Arguments	There is one column each of predicate argument structure information for the predicate mentioned in Column 7.
N	Coreference	Coreference chain information encoded in a parenthesis structure.

Figure A.1: Information contained in each column in the CoNLL-2012 data (Pradhan, Moschitti, et al., 2012).

```
bc/cctv/00/cctv_0001   0    0          The     DT     (TOP(S(NP*        -    -   -   Speaker#1   (EVENT*   (ARG1*           *   (65
bc/cctv/00/cctv_0001   0    1      Hundred    NNP          (NML*        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0    2   Regiments   NNPS             *)        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0    3   Offensive    NNP             *)        -    -   -   Speaker#1        *)       *)          *    65)
bc/cctv/00/cctv_0001   0    4         was    VBD          (VP*        be   01   1   Speaker#1        *      (V*)          *    -
bc/cctv/00/cctv_0001   0    5         the     DT     (NP(NP(NP*        -    -   -   Speaker#1        *   (ARG2*       (ARG1*    -
bc/cctv/00/cctv_0001   0    6    campaign     NN             *)        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0    7          of     IN          (PP*        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0    8         the     DT          (NP*        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0    9     largest    JJS             *        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0   10       scale     NN           *)))       -    -   -   Speaker#1        *        *           *)   -
bc/cctv/00/cctv_0001   0   11   launched     VBN          (VP*    launch   01   2   Speaker#1        *        *         (V*)   -
bc/cctv/00/cctv_0001   0   12          by     IN          (PP*        -    -   -   Speaker#1        *        *       (ARG0*    -
bc/cctv/00/cctv_0001   0   13         the     DT          (NP*        -    -   -   Speaker#1     (ORG*       *           *   (36
bc/cctv/00/cctv_0001   0   14      Eighth    NNP          (NML*        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0   15       Route    NNP             *)        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0   16        Army    NNP            *))        -    -   -   Speaker#1        *)       *           *)   36)
bc/cctv/00/cctv_0001   0   17      during     IN          (PP*        -    -   -   Speaker#1        *        *   (ARGM-TMP*    -
bc/cctv/00/cctv_0001   0   18         the     DT       (NP(NP*        -    -   -   Speaker#1   (EVENT*       *           *    -
bc/cctv/00/cctv_0001   0   19         War    NNP             *)        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0   20          of     IN          (PP*        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0   21  Resistance    NNP       (NP(NP*)       -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0   22     against     IN          (PP*        -    -   -   Speaker#1        *        *           *    -
bc/cctv/00/cctv_0001   0   23       Japan    NNP   (NP*))))))))       -    -   -   Speaker#1        *)       *)          *)   -
bc/cctv/00/cctv_0001   0   24           .      .            *))       -    -   -   Speaker#1        *        *           *    -
```

Figure A.2: An example of an annotated sentence from the CoNLL-2012 data (English).

Field #	Name	Description
1	ID	Token counter, starting at 1 for each new sentence
2	FORM	Form or punctuation symbol (the token; “split” for English)
3	LEMMA	Gold-standard lemma of FORM
4	PLEMMA	Automatically predicted lemma of FORM
5	POS	Gold-standard POS (major POS only)
6	PPOS	Automatically predicted major POS by a language-specific tagger
7	FEAT	Gold-standard morphological features (if applicable)
8	PFEAT	Automatically predicted morphological features (if applicable)
9	HEAD	Gold-standard syntactic head of the current token (ID or 0 if root)
10	PHEAD	Automatically predicted syntactic head
11	DEPREL	Gold-standard syntactic dependency relation (to HEAD)
12	PDEPREL	Automatically predicted dependency relation to PHEAD
13	FILLPRED	Contains ‘Y’ for argument-bearing tokens
14	PRED	(sense) identifier of a semantic “predicate” coming from a current token
15...	APREDn	Columns with argument labels for each semantic predicate (in the ID order)

Figure A.3: Information contained in each column in the CoNLL-2009 data (Hajič et al., 2009).

1	建筑	建筑	建筑	NN	NN	_	_	2	NMOD	_	_	_	_	_
2	公司	公司	公司	NN	NN	_	_	3	SBJ	_	A0	_	_	_
3	进	进	进	VV	VV	_	_	0	ROOT	进.01	_	_	_	_
4	区	区	区	NN	NN	_	_	3	COMP	_	A1	_	_	_
5	,	,	,	PU	PU	_	_	3	CJTN	_	_	_	_	_
6	有关	有关	有关	JJ	JJ	_	_	7	AMOD	_	_	_	_	_
7	部门	部门	部门	NN	NN	_	_	9	SBJ	_	_	A0	_	_
8	先	先	先	AD	AD	_	_	9	ADV	_	_	ADV	_	_
9	送上	送上	送上	VV	VV	_	_	5	CJT	送上.01	_	_	_	_
10	这些	这些	这些	DT	DT	_	_	12	DMOD	_	_	_	_	_
11	法规性	法规性	法规性	NN	JJ	_	_	12	NMOD	_	_	_	_	_
12	文件	文件	文件	NN	NN	_	_	9	COMP	_	_	A1	_	_
13	,	,	,	PU	PU	_	_	3	cCJTN	_	_	_	_	_
14	然后	然后	然后	AD	AD	_	_	15	ADV	_	_	_	ADV	_
15	有	有	有	VE	VE	_	_	13	CJT	有.05	_	_	_	_
16	专门	专门	专门	JJ	JJ	_	_	17	AMOD	_	_	_	_	_
17	队伍	队伍	队伍	NN	NN	_	_	18	SBJ	_	_	_	_	A0
18	进行	进行	进行	VV	VV	_	_	15	COMP	进行.01	_	_	A1	_
19	监督	监督	监督	NN	NN	_	_	20	NMOD	_	_	_	_	_
20	检查	检查	检查	NN	NN	_	_	18	COMP	_	_	_	_	A1
21	。	。	。	PU	PU	_	_	3	UNK	_	_	_	_	_

Figure A.4: An example of an annotated sentence from the CoNLL-2009 data (Chinese).

1	Les	15	15	DET	DET	_	_	2	2	det	det	_	_	_	
2	services		368	368	NC	NC	_	_	4	4	suj	suj	_	_	A1
3	postaux	0	0	ADJ	ADJ	_	_	2	2	mod	mod	_	_	_	
4	doivent	186	186	V	V	_	_	0	0	root	root	_	_	AM-MOD	
5	rester	315	315	VINF	VINF	_	_	4	4	obj	obj	_	remain.01		_
6	des	9	9	P+D	P+D	_	_	5	5	de_obj	de_obj	_	_	A3	
7	services		368	368	NC	NC	_	_	6	6	obj	obj	_	_	_
8	publics	204	204	ADJ	ADJ	_	_	7	7	mod	mod	_	_	_	
9	.	2	2	PONCT	PONCT	_	_	4	4	ponct	ponct	_	_	_	

Figure A.5: An example of an annotated sentence from the Van der Plas, Merlo, et al. (2011) data in CoNLL-2009 format (French).

Appendix B

Chapter 4.4

```
{
  "dataset_reader":{"type":"srl"},
  "train_data_path": "./Data/conll-formatted-ontonotes-5.0/data/train",
  "validation_data_path": "./Data/conll-formatted-ontonotes-5.0/data/development",
  "model": {
    "type": "srl",
    "text_field_embedder": {
      "token_embedders": {
        "tokens": {
            "type": "embedding",
            "embedding_dim": 300,
            "pretrained_file": "./Embeddings/wiki.aligned.enfrzh.vec.tar.gz",
            "trainable": true
        }
      }
    },
    "initializer": [
      [
        "tag_projection_layer.*weight",
        {
          "type": "orthogonal"
        }
      ]
    ],
    "encoder": {
      "type": "alternating_lstm",
      "input_size": 400,
// NOTE: 100 dim for predicate location + 300 embeddings size
      "hidden_size": 300,
      "num_layers": 8,
      "recurrent_dropout_probability": 0.1,
      "use_highway": true
    },
    "binary_feature_dim": 100
  },
  "iterator": {
    "type": "bucket",
    "sorting_keys": [["tokens", "num_tokens"]],
    "batch_size" : 80
  },

  "trainer": {
    "num_epochs": 500,
    "grad_clipping": 1.0,
    "patience": 20,
    "validation_metric": "+f1-measure-overall",
    "cuda_device": 0,
    "optimizer": {
      "type": "adadelta",
      "rho": 0.95
    }
  }
}
```

Figure B.1: An example of a configuration training file in jsonnet format.