NEURAL NAME TAGGING
FOR LOW-RESOURCE LANGUAGES

Boliang Zhang

Submitted in Partial Fullfillment of the Requirements
for the Degree of

DOCTOR OF PHILOSOPHY

Approved by:
Heng Ji, Chair
Kyunghyun Cho
James A. Hendler
Deborah L. McGuinness
Luke Zettlemoyer

Department of Computer Science
Rensselaer Polytechnic Institute
Troy, New York

May 2019
(For Graduation May 2019)

Contents

List of Tables vi

List of Figures vii

ACKNOWLEDGMENT ix

ABSTRACT xii

1. Introduction 1

1.1 Information Extraction and Name Tagging 1

1.2 Low-resource Languages 3

1.3 Hypothesis and Solutions 4

1.4 Task Definition 10

1.4.1 Problem Formulation 10

1.4.2 Evaluation 11

1.5 Contributions of the Thesis 12

1.6 Related Publications 13

2. Related Work 14

2.1 Traditional Name Tagging 14

2.2 Neural Name Tagging 16

2.3 Multilingual and Low-resource Language Name Tagging 18

3. Noisy Annotation Acquisition 20

3.1 “Chinese Room” 20

3.2 Cross-lingual Name Projection 21

3.3 Wikipedia Knowledge Base (KB) mining 21

4. Incorporating Linguistic resources: Expectation-driven Learning 23

4.1 Time Zero: Language Universals 23

4.2 Expectation Learning 24

4.2.1 Approach Overview 24

4.2.2 Survey with Native Speaker 24

4.2.3 Mono-lingual Expectation Mining 25

4.2.4 Cross-lingual Expectation Projection 26

4.3 Supervised Active Learning 26
4.4 Cost-aware Combination 27
4.5 Experiments 28
4.5.1 Data 28
4.5.2 Cost-aware Overall Performance 29
4.5.3 Comparison of Expectation Discovery Methods 29
4.5.4 Impact of Supervised Active Learning 31
5. Incorporating Linguistic resources: Non-traditional Linguistic Resources 33
5.1 Non-traditional Linguistic Resources 33
5.2 Approach Overview 34
5.2.1 Baseline's Sensitiveness to Noise 34
5.2.2 A New Improved Model 34
5.3 Incorporating Non-traditional Linguistic Knowledge 35
5.3.1 Basic Knowledge about the Language 35
5.3.2 Linguistic Structure 36
5.3.3 Multi-lingual Dictionaries 37
5.3.4 Encoding Linguistic Features 40
5.4 Experiments 40
6. Global Attention for Name Tagging 43
6.1 Approach Overview 43
6.1.1 Document-level Attention 43
6.1.2 Topic-aware Corpus-level Attention 45
6.1.3 Tag Prediction 46
6.2 Experiments 46
6.2.1 Dataset 46
6.2.2 Experimental Setup 47
6.2.3 Performance Comparison 48
6.2.4 Qualitative Analysis 50
7. Cross-lingual Language Model for Name Tagging 53
7.1 Approach Overview 53
7.1.1 Cross-lingual Word Embeddings 53
7.1.2 Cross-lingual Language Model 54
7.2 Experiments 56

7.2.1 Dataset 56
7.2.2 Training Details 56
7.2.3 Model Comparison 57
8. System Demo 59
8.1 System Overview 59
8.2 Entity Extraction 59
8.3 Entity Linking and Localization 61
8.4 Training and Testing APIs 62
8.5 Testing Interface 63
8.6 Heatmap Visualization 63
9. Conclusions and Future Work 65
9.1 Conclusions 65
9.2 Future Work 67
9.2.1 Enhanced Cross-lingual Contextualized Word Embedding 67
9.2.2 Unsupervised Cross-lingual Information Extraction 68
9.2.3 Joint Information Extraction 69
9.3 Final Remark 69
REFERENCES 70

List of Tables

1.1 A list of generic entity types. 2

1.2 Named entity tagging as a sequence model, showing BIO and BIOES schemas. 11

2.1 Typical features for a feature-based name tagging system. 15

2.2 An example of name tagging features for an English sentence. 15

4.1 Universal Name Tagger Features 23

4.2 Survey Question Examples 25

4.3 Data Statistics 28

4.4 Contributions of Various Expectation Discovery Methods (F-score %) 31

5.1 Name-related Knowledge Summarized from Grammar Books. 37

5.2 Name-related Knowledge Extracted from WALS. 38

5.3 Common Syllable Patterns Extracted from CIA Names. 38

5.4 Name Related List Statistics (# of entries). 40

5.5 Feature Integration Methods Comparison. 40

5.6 Contributions of Various Categories of Linguistic Knowledge (F-score (%)). . . 42

6.1 # of tokens in name tagging datasets statistics. # of names is given in parentheses. 47

6.2 Hyper-parameters. 48

6.3 Performance of our methods versus the baseline and state-of-the-art models. . 52

7.1 # of sentences in name tagging datasets statistics. 56

7.2 Hyper-parameters and training setup. 57

7.3 Model comparisons for English-Russian and English-Spanish. 57

8.1 Runtime APIs description. 59

8.2 Training APIs description. 60

8.3 Name Tagging Performance on Low-Resource Languages 62

List of Figures

1.1 An example of name tagging on English sentences. 2

1.2 Example from the baseline and our model with some supporting evidence. . . . 6

1.3 A toy example of aligning cross-lingual word embeddings. 8

1.4 A toy example of projecting cross-lingual language model into a shared space. . 9

2.1 A toy word embedding example. 16

2.2 Explanation of word embedding composition: $w_{KING} - w_{MAN} + w_{WOMAN} = w_{QUEEN}$. 17

2.3 Word level Bidirectional LSTM name tagging architecture. 18

2.4 Word+character level Bidirectional LSTM name tagging architecture. 19

3.1 Noisy Training Data Generation by Projecting English Automatic Name Annotations to Hausa. 21

4.1 Expectation Driven Name Tagger Overview 24

4.2 Cost-aware Expectation Composition 28

4.3 Comparison of methods combining expectation-driven learning and supervised active learning given various time bounds 30

4.4 Hausa Supervised Active Learning Curve 31

4.5 Active Learning vs. Passive Learning (%) 32

5.1 Performance of baseline DNN Name Taggers Trained from Data with Various Noise Levels (The noise level is created by assigning the proportion of $Train_{noise}$ in $Train_{mix}$ as 0%, 25%, 50%, 75% and 100% respectively.) 35

5.2 Three Integration Methods to Incorporate Explicit Linguistic Features into DNN. 36

5.3 Name Tagging Performance. 41

5.4 Examples of Corrections Made by Each Category of Linguistic Knowledge. . . 42

6.1 Document-level Attention Architecture. (Within-sequence context in red incorrectly indicates the name as PER, and document-level context in green correctly indicates the name as ORG.) 43

6.2 Corpus-level Attention Architecture. 46

6.3 Average F1 score for each epoch of the ten runs of our model with both document-level and corpus-level attentions. Epochs 1-50 are the pre-training phase and 51-100 are the fine-tuning phase. 48

6.4 Comparison of name tagging results between the baseline and our methods. . . 50

7.1 An example of training HL and LL language model. 54

7.2 An example of back-translation. 55

7.3 Learning curve of finetuning pre-trained HL name tagger on LL annotations. . 58

8.1 Cross-lingual Entity Extraction and Linking Interface 60

8.2 Cross-lingual Entity Extraction and Linking Testing Result Visualization . . . 60

8.3 Heatmap Visualization 61

8.4 Different Map Styles 64

ACKNOWLEDGMENT

The past six years at RPI have been an unforgettable and invaluable experience to me. I cannot express my gratitude enough for those individuals who have advised, supported, and influenced me during my journey of graduate study.

My greatest thanks go to my advisor, Heng Ji. She is a great mentor. At the time I came to RPI, I was a master student, knew little about research and never heard of Natural Language Processing (NLP). At the time I joined BLENDER lab, Heng provided me very constructive advice, from fundamental principles of doing research to high-level applications of NLP. She is a hands-on type of adviser who always provides help on paper writing and experiment implementation. Heng is a role model and diligent person who dedicates life to science, and pursues perfection and excellence. Her work style deeply influenced me during my graduate study and will be a beacon that lights the path in my future career.

I would like to thank my committee members: Kyunghyun Cho, James A. Hendler, Deborah L. McGuinness and Luke Zettlemoyer for their great support and constructive advice during my proposal and thesis composing. Kyunghyun Cho is one of the pioneers in the field of Deep Learning. Many of the ideas in this thesis are inspired or built on top of his famous paper on neural attention. James and Deborah are the two giants of the field of Semantic web. Their early foreseeing contributions to the field of Artificial Intelligence still have great impact in these days. They provided deep thoughts and constructive comments on my proposal and thesis. Luke Zettlemoyer initiated a new research direction in NLP by his paper that first uses pre-training contextualized word embeddings for various NLP tasks. I learned a lot from his papers.

It is my great honor to work with Kevin Knight in the DARPA LORELEI project which I have been working on for almost the entire time of my PhD study. Kevin Knight is one of the pioneers in the field of Machine Translation. His perfectionism influenced every team member and led us to success, one after another. I look forward to continuing working with him in the near future.

During my PhD, I have done four wonderful internships at Microsoft Research Aisa, Interactions LLC, Facebook and DiDi Chuxing. I thank my mentor Chin-Yew Lin, John Chen and Yang Liu when I worked at these places.

Research is never done in a vacuum. I'm deeply indebted to my colleagues and co-authors. I thank Xiaoman Pan for our seamless collaboration on many papers and projects. I thank my fellow collaborators: Lifu Huang, Spencer Whitehead, Hongzhao Huang, Di Lu, Tongtao Zhang, Ying Lin, Qingyun Wang, Zhiying Jiang, and others.

This thesis was supported by the U.S. DARPA LORELEI Program No. HR0011-15-C-0115 and ARL/ARO MURI W911NF-10-1-0533. I thank them for financially supporting me.

My last but the warmest thanks to my parents: Hongwei Wang and Chenqiao Zhang. I thank them for giving me the best education, and always respecting my decisions. I could not get where I am without their support.

To my parents: Hongwei Wang and Chenqiao Zhang,
for their unconditional love.

ABSTRACT

Extracting information from natural language text is one of the most challenging and long-standing problems in the field of Natural Language Processing (NLP). Information Extraction (IE) turns the unstructured data into a structured knowledge base, according to a predefined schema or ontology. One of the core tasks in Information Extraction is name tagging, which seeks to identify and classify names in text into predefined categories such as persons, locations and organizations. It is also known as Named Entity Recognition (NER). Name tagging produces informative results that are beneficial for many downstream NLP tasks, such as relation extraction and event extraction, and it also plays an important role in industrial applications, such as Question Answering and Dialogue System.

State-of-the-art name tagging approaches rely on supervised machine learning models that require a massive amount of clean annotated data. These supervised methods are sophisticated and very effective for high-resource languages (HL) such as English, German, and French. However, in scenarios where annotations are insufficient and noisy, the performance of these approaches declines greatly. Meanwhile, the acquisition of human annotated data is expensive and time-consuming, which makes traditional supervised machine learning approaches very difficult to deploy, especially in an emergent setting.

In this thesis, we focus on tackling the challenges of name tagging for low-resource languages (LL) in emergent situations. The methodology presented in this thesis consists of three parts. In the first part, we populate name tagging annotations by generating "silver-standard" noisy training data via 1) *"Chinese Room"* where we designed a "Chinese Room" [1] platform to ask a native English speaker to extract names from some low-resource language documents, 2) *Parallel Name Projection* where we project extracted English names to LL sentences through English-LL parallel data, and (3) *Wikipedia Knowledge Base (KB) mining* where we transfer annotations from English to other languages through cross-lingual links and KB properties in Wikipedia.

In the second part, as the traditional supervised machine learning models suffer from a huge performance decrease when trained on the noisy "silver-standard" annotations, we propose a new solution to incorporate many *non-traditional* language universal resources that are readily available but rarely explored in the NLP community. These universal resources contain valuable dictionaries, grammars, language patterns, etc, all of which are presented in multiple languages. We encode various types of non-traditional linguistic resources as features into a supervised Deep Neural Network (DNN) name tagger.

In the third part, as only relying on local contextual information, the current DNN models may perform poorly when the local context is ambiguous or limited. We propose a new framework to improve the DNN name tagger by utilizing local and global (document-level and corpus-level) contextual information. We retrieve the document-level context from other sentences within the same document and corpus-level context from sentences in other documents. The proposed model learns to incorporate document-level and corpus-level contextual information alongside local contextual information via global attention, which dynamically weights their respective contextual information, and gating mechanisms, which determine the influence of this information.

At last, we investigate training LL name taggers without using any LL annotation. We transfer a name tagger that trained on HL annotations to a LL name tagger via two unsupervised approaches: 1) cross-lingual word embedding where we align monolingual word embedding of HL and LL into a shared space, and 2) cross-lingual language model where instead of aligning word embedding, we project the contextualized word embedding (language model) of HL and LL into a shared space.

Chapter 1
Introduction

1.1 Information Extraction and Name Tagging

Information Extraction (IE) turns the unstructured information embedded in texts into structured data [2], according to a predefined schema or ontology. The transformed structured information are sets of machine-readable facts that can be further populated into a knowledge base. Rich structured information distilled from enormous unstructured data largely expands knowledge bases [3, 4] and significantly improves downstream Natural Language Processing (NLP) tasks, such as question answering [5, 6, 7] and machine translation [8].

Name Tagging - The initial step in most IE tasks is to identify and classify mentions of named entities in a text into certain entity types. The entity types are predefined and task-specific: person, location, and organization are common types in news domain, but entity types such as protein and gene are common in bio-medical domain, and currency is a common entity type in finance domain.

Relation Extraction - Once names are extracted from texts, we can apply relation extraction to discover and classify the relations among the entities.

Event Extraction - Event extraction extracts the specific knowledge of certain incidents in which these entities participate.

Besides these three tasks, IE also includes coreference resolution, entity linking, slot filling, etc. In this thesis, we mainly focus on the task of name tagging.

Name Tagging is a process where an algorithm takes a string of text (sentence or paragraph) as input and identifies relevant proper nouns that are mentioned in the string. Conceptually it can be broke down as two distinct phases: named entity detection and entity type classification. The first phase is commonly simplified as a segmentation or chunking problem where names are defined as continuous spans of words, which means a name is a single word or phrase. In most cases, nested names are disallowed, for instance, “Illinois State Senate” is an organization name as a whole, regardless of the fact that its substring “Illinois” is itself a name. The second phase, entity type classification, assigns each identified name span a predefined type.

Type	Tag	Sample Categories	Example Sentences
Person	PER	people, social media usernames	[**PER Barack Obama**] was born on August 4, 1961.
Location	LOC	rivers, mountains, oceans, roads	The [**LOC Grand Canyon**] is a steep-sided canyon carved by the [Colorado River LOC].
Geo-political Entity	GPE	countries, cities, states, provinces	[**GPE New York City**] is the most populous city in the [**GPE United States**].
Organization	ORG	companies, military bases, sports teams	[**ORG Apple**] is the first US company worth $1 trillion.
Facility	FAC	airports, bridges, buildings	The [**FAC Golden Gate Bridge**] is a suspension bridge.
Vehicles	VEH	airplanes, ships, trains	The disappearance of [**VEH MH370**] has been dubbed one of the greatest aviation mysteries of all time.
Weapons	WEA	tanks, missiles	The [**WEA Patriot**] is a ground-based, mobile missile defense interceptor.

Table 1.1: A list of generic entity types.

Table 1.1 lists typical generic named entity types. Name tagging is also known as Named Entity Recognition (NER) in some literature.

[PER Barack Hussein Obama II] (born August 4, 1961) is an **[GPE American]** attorney and politician who served as the 44th President of the **[GPE United States]** from January 20, 2009, to January 20, 2017. A member of the **[ORG Democratic Party]**, he was the first African American to serve as president. **[PER Barack Obama]** was previously a **[GPE United States]** Senator from **[GPE Illinois]** and a member of the **[ORG Illinois State Senate]**.

Figure 1.1: An example of name tagging on English sentences.

Figure 1.1 shows a name tagging example. The text contains 8 mentions of named entities, including two Persons (PER), four Geo-Political Entities (GPE), and two Organizations (ORG). Ambiguous cases are underlined, as a word/phase is tagged as a name only when it refers to a specific and unique entity, while in this example, "President" is only a title and "African American" refers to a group of people.

Name tagging has many applications. In Question Answering (QA), incorporating name tagging improves the speed and accuracy of getting correct answers [9, 6, 10]. In

Machine Translation (MT), the extraction of named entities beforehand largely improves translation results [8].

1.2 Low-resource Languages

There are about 11,000 human languages in the world, depending on different categorization criteria. The most spoken language is English in terms of regions, and if in terms of population, Chinese is the language that has the largest number of native speakers. Other widespreadly used languages include European languages, such as French, Portuguese, and Spanish, and United Nation (UN) languages, such as Arabic and Russian. These languages are considered as high-resource languages (HL) as they have an enormous number of monolingual text from either literature or internet, as well as a large amount of publicly available human annotations that created for various NLP tasks, e.g. name tagging and machine translation. In contrast, most of the 11,000 languages can be considered as low-resource languages (LL). Formal definition of low-resource language is difficult to pin down, but most of LL come with the following characteristics [11]: *low resource/low density* which refers to languages "for which few online resources exist" [12] or "for which few computational data resources exist" [13], *critical condition* which has typically referred to languages that suffer an undesirable ratio of supply to demand, and *endangered condition* which refers to languages that are at risk of losing their native speakers through a combination of death and shift to other languages.

Studying low-resource languages can make huge impacts on real world missions or applications. To cope with humanitarian challenges such as disease outbreaks and natural calamities, government rescue resources are usually deployed, typically in regions of the world where one or more low-resource languages are frequently used in formal or informal media. In such situations, information tools, such as information extraction and machine translation for low-resource languages are necessary and in great demand.

In the internet era, the world is "online". People from different regions communicate with each other on different languages. For social good reasons, studying low-resource languages can largely reduce the barriers between users that speaking different languages and make the communication more efficient. The potential business value involved cannot be neglected as well.

1.3 Hypothesis and Solutions

In this section, we present four hypotheses to address the challenges and overcome the limitations discussed above. The approaches proposed in this thesis are originated from these hypotheses as well.

Hypothesis 1. Noisy low-resource language annotations, that created by human efforts from non-native speakers, or distilled automatically from existing resources, can provide supervisions to machine learning models.

In the preliminary step of our proposed approach, we automatically generate noisy labeled data by the following three methods:

"Chinese Room" - When we are in a foreign country, even if we don't know the language, we would still be able to guess the word "*gate*" from the airport broadcast based on its frequency and position in a sentence, and guess the word "*station*" by pattern mining of many subway station labels. We design a "*Chinese Room*" [1] platform by asking a human user who are not native low-resource language (LL) speaker to annotate names in LL text.

Cross-lingual Name Projection through Parallel Data - When IL-English parallel data is available, we apply a state-of-the-art English name tagger to the English documents to obtain a list of expected names. Then we translate the English patterns and expected names to IL. When there is no human constructed English-to-IL lexicon available, we derive a word-for-word translation table from a small parallel data set using the GIZA++ word alignment tool [14]. We also convert IL text to Latin characters based on Unicode mapping,[1] and then apply Soundex code [15, 16] to find the IL name equivalent that shares the most similar pronunciation as each English name. For example, the Bengali name "টনি ব্লেয়ার" and "*Tony Blair*" have the same Soundex code "*T500 B460*".

Wikipedia Knowledge Base (KB) Mining - Wikipedia is an enormously multilingual resource that currently houses 301 languages. It contains naturally annotated markups and rich information structures through crowd-sourcing for 35 million articles in 3 billion words. Name mentions in Wikipedia are often labeled as anchor links to their corresponding referent pages. We leverage these anchor links for developing a language universal framework to automatically extract name mentions from Wikipedia articles [17].

Hypothesis 2. Language universal features can alleviate the impact of noise in annotations, and provide robustness and generalization to a weakly

[1] http://www.ssec.wisc.edu/ tomw/java/unicode.html

supervised machine learning model.

The automatically acquired annotations through aforementioned approaches contain various kinds of noise, such as missing error where a name is not labeled, spurious error where a labeled phrase should not be a name, offset error where only part of a name is labeled. Various types of noise hurt the traditional supervised model by a decrease of 20%-30% on performance based on our experiments.

In order to improve the robustness of name tagging to noise, we propose to exploit a wide variety of multi-lingual resources, such as World Atlas of Linguistic Structure (WALS) [18], Central Intelligence Agency (CIA) Names, grammar books, and survival guides. Such resources have been largely ignored by the mainstream statistical NLP research, because they were not specifically designed for NLP purpose at the first place and they are often far from complete. Thus they are not immediately actionable - converted into features, rules or patterns for a target NLP application. We design various methods to convert them into machine readable features for a new DNN architecture.

Hypothesis 3. Recognizing names not only relies on the context of the sentence, but also the context of the article, sometimes even the whole corpus.

When labeling a token, local context (*i.e.,* surrounding tokens) is crucial because the context gives insight to the semantic meaning of the token. However, there are many instances in which the local context is ambiguous or lacks sufficient content, especially when the target LL lacks of linguistic resources and tools. Figure 1.2 shows an example in English. The query sentence discusses "`Zywiec`" selling a product and profiting from these sales, but the local contextual information is ambiguous as more than one entity type could be involved in a sale. As a result, the baseline model mistakenly tags "`Zywiec`" as a person (PER) instead of the correct tag, which is organization (ORG). If the model has access to supporting evidence that provides additional, clearer contextual information, then the model may use this information to correct the mistake given the ambiguous local context.

Additional context may be found from other sentences in the same document as the query sentence (**document-level**). In Figure 1.2, the sentences in the document-level supporting evidence provide clearer clues to tag "`Zywiec`" as ORG, such as the references to "`Zywiec`" as a "`firm`".

In cases where the sentences at the document-level cannot serve as a source of additional context, one may find additional context from sentences in other documents in the corpus

Baseline:

So far this year [PER Zywiec], whose full name is Zaklady Piwowarskie w Zywcu SA, has netted six million zlotys on sales of 224 million zlotys.

Our model (Document-level + Corpus-level Attention):

So far this year [ORG Zywiec], whose full name is Zaklady Piwowarskie w Zywcu SA, has netted six million zlotys on sales of 224 million zlotys.

Document-level Supporting Evidence:

> Van Boxmeer also said [ORG Zywiec] would be boosted by its recent shedding of soft drinks which only accounted for about three percent of the firm's overall sales and for which 7.6 million zlotys in provisions had already been made.

> Polish brewer [ORG Zywiec]'s 1996 profit slump may last into next year due in part to hefty depreciation charges, but recent high investment should help the firm defend its 10-percent market share, the firm's chief executive said.

Corpus-level Supporting Evidence:

> The [ORG Zywiec] logo includes all of the most important historical symbols of the brewery and Poland itself.

> [LOC Zywiec] is a town in south-central Poland 32,242 inhabitants (as of November 2007).

Figure 1.2: Example from the baseline and our model with some supporting evidence.

(**corpus-level**). Figure 1.2 shows some of the corpus-level supporting evidence for "Zywiec". In this example, similar to the document-level supporting evidence, the first sentence in this corpus-level evidence discusses the branding of "Zywiec", corroborating the ORG tag. Whereas the second sentence introduces noise because it has a different topic than the current sentence and discusses the Polish town named "Zywiec", one may filter these noisy contexts, especially when the noisy contexts are accompanied by clear contexts like the first sentence.

We propose to utilize local, document-level, and corpus-level contextual information to improve name tagging. Generally, we follow the *one sense per discourse* hypothesis introduced by [19]. Some previous name tagging efforts apply this hypothesis to conduct majority voting for multiple mentions with the same name string in a discourse through a cache model [20] or post-processing [21]. However, these rule-based methods require manual tuning of thresholds. Moreover, it's challenging to explicitly define the scope of discourse. We propose a new neural network framework with global attention to tackle these challenges. Specifically, for each token in a query sentence, we propose to retrieve sentences that contain the same token from document-level and corpus-level contexts (*e.g.,* document-level and corpus-level supporting evidence for "`Zywiec`" in Figure 1.2). To utilize this additional information, we propose a model that first produces representations for each token that encode the local context from the query sentence as well as the document-level and corpus-level contexts from the retrieved sentences. Our model uses a *document-level attention* and *corpus-level attention* to dynamically weight the document-level and corpus-level contextual representations, emphasizing the contextual information from each level that is most relevant to the local context and filtering noise such as the irrelevant information from the mention "`[LOC Zywiec]`" in Figure 1.2. The model learns to balance the influence of the local, document-level, and corpus-level contextual representations via gating mechanisms. Our model predicts a tag using the local, gated-attentive document-level, and gated-attentive corpus-level contextual representations, which allows our model to predict the correct tag, ORG, for "`Zywiec`" in Figure 1.2.

Hypothesis 4. Distributed word embeddings/contextualized word embeddings of HL and LL can be projected into a shared space, so that machine learning models trained on HL embeddings/contextualized embeddings can produce satisfying performance on LL.

Cross-lingual Word Embeddings - Conceptually, word embeddings are distributional representations that aim to quantify and categorize semantic similarities between words. They are unsupervisedly trained from massive monolingual text data. Word embeddings are essential components of neural networks. A typical word-level neural model for name tagging takes word embeddings as input, followed by a Recurrent Neural Network (RNN) over each word to incorporate contextual information, and at the end, a feedforward network plus a softmax or Conditional Random Field (CRF) layer predicts the labels. Word embeddings are the crucial features that neural models rely on when making prediction.

If there exists a vector space where the structures of word embeddings from two languages are highly overlapped, a name tagging model trained on one language can be directly adapted to the other language as the word embedding features used for prediction are shared

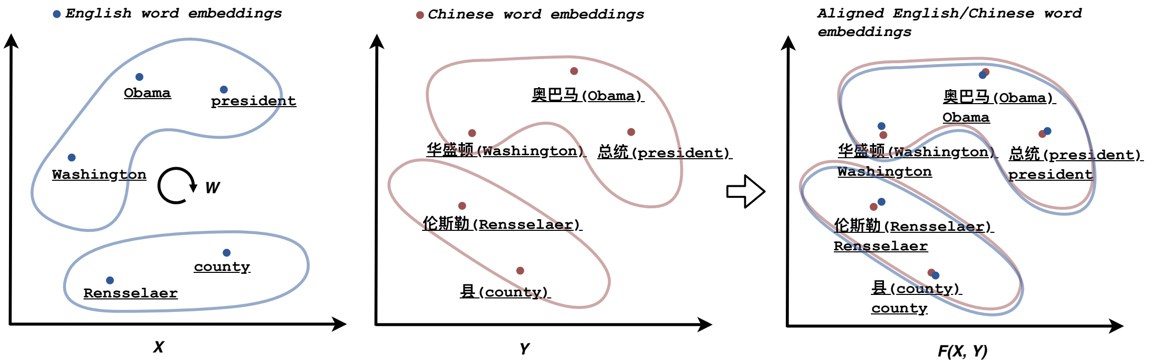


Figure 1.3: A toy example of aligning cross-lingual word embeddings.

between the two languages. For a concrete example, in a Russian sentence "Путин уехал в Москву (Putin went to Moscow)", when Russian and English word embeddings are perfectly aligned in a shared space, the closest English word embeddings to the Russian words are "Putin", "went", "to" and "Moscow". An English name tagger that classifies "Putin" as PER and "Moscow" as GPE is able to classify "Путин" as PER and "Москву" as GPE.

[22] first observes that continuous word embedding spaces exhibit similar structures across languages, even when considering distant language pairs such as English and Russian. To exploit such similarity, [23, 24, 25] propose to learn a linear mapping between the source and target word embedding spaces. The linear mapping is learned upon a bilingual dictionary where words pairs are used as anchor points. They evaluate the linear mapping quality via a word translation task. Recently, [26] proposes an adversarial approach that does not require any bilingual dictionary to unsupervisedly learn the mapping. Figure 1.3 presents the process of linearly aligning word embeddings of English and Chinese.

In our experiments, we utilize the MUSE toolkit that is published in [26] to unsupervisedly align word embeddings of HL and LL. We train a Bi-LSTM CRF name tagger on HL labeled data by initializing its word embeddings with the aligned cross-lingual MUSE embeddings during training. Then we directly adapt the HL name tagger to LL sentences by taking the LL portion of cross-lingual embeddings as input. In the experiments, we consider English as HL and Russian/Spanish as LL. Without any Russian/Spanish labeled data, the English name tagger trained on cross-lingual word embeddings achieved **37.81%** F-1 on Russian (LRLP), **50.13%** on Spanish (CoNLL 2003), compared to the state-of-the-art performance using Russian/Spanish labeled training data: 57.66% and 83.31%. Moreover, when annotations for LL are available, we initialize an LL name tagger with the English name tagger model parameters and finetune it on LL labeled data. The finetuned model achieved **65.17%** and **83.61%** on Russian and Spanish, which outperforms the state-of-the-art.

Cross-lingual Contextualized Word Embeddings - As trained on monolingual corpus, fundamentally word embeddings only present the semantic meaning of words in that corpus, but when given a context, word embeddings cannot represent the contextual information. Contextualized word embedding is proposed to capture both semantic meanings and contextual information of words in a sentence. As pre-training contextualized word embedding is similar to training language model, the concept of contextualized word embeddings and language model are sometimes interchangeable in literature. Contextualized word embedding pre-training has shown to be effective for improving many NLP tasks [27, 28, 29, 30, 31, 32]. For the problem of name tagging, [30] first integrated a multi-layer Bi-LSTM pre-trained language model to a Bi-LSTM CRF name tagger and significantly outperforms the state-of-the-art. [31] uses a forward only transformer [33] pre-trained language model and [27] uses a bi-directional transformer pre-trained language model to improve name tagger performance. Especially [27] completely remove pre-trained word embeddings and only feed pre-trained language model to a Bi-LSTM name tagging model and achieve a new state-of-the-art performance. This observation somewhat proves pre-trained language model produces better word representations over word embeddings.

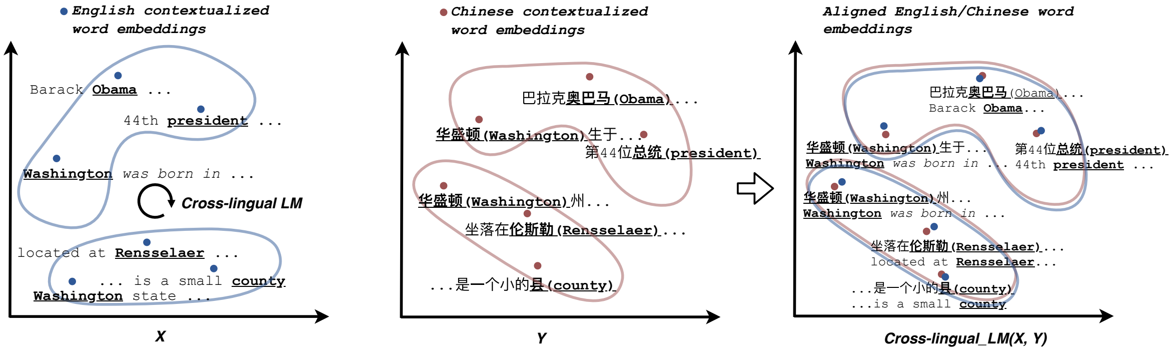


Figure 1.4: A toy example of projecting cross-lingual language model into a shared space.

Inspired by the idea of aligning word embeddings, we investigate projecting the contextualized word embeddings (language model) of HL and LL into a shared space, so as to further improve the language transferring capability of neural name taggers from HL to LL. Pre-training cross-lingual word embeddings is essentially the process of learning a bilingual word translation, while pre-training cross-lingual language model aims to learn word translation not only based on the word itself but also the context, which can largely eliminate word ambiguity. For example, for an English phrase “apple operating system”, English-Spanish cross-lingual word embeddings may produce word translations (“manzana”, “operativo”, “sistema”) in which “manzana” refers to the fruit apple which is incorrect in this case. While the cross-lingual contextualized word embedding of “apple” in this phrase is positioned in the cluster of companies because of the context, and ideally it can be mapped to the company Apple.

Our goal is to unsupervisedly pre-train cross-lingual contextualized word embeddings from HL and LL monolingual corpora. [34] demonstrates the possibility of using “back-translation” to unsupervisedly train a machine translation model. Following the same idea, we use “back-translation” as the bridge to pre-train corss-lingual contextualized word embeddings. Experiments show that cross-lingual contextualized word embeddings significantly outperforms cross-lingual word embeddings in name tagging language transferring task.

1.4 Task Definition

1.4.1 Problem Formulation

The most successful methods for name tagging are based on supervised learning, by modeling name tagging as a sequence labeling problem. A tagging schema assigns each word of the sentence a label that indicates its position in a name span and the type of the name. The most commonly used schema is BIO where B stands for **B**egin, I stands for **I**nside, and O stands for **O**utside. A more sophisticated tagging schema is BIOES(U) that distinguishes between the end of a named entity and single entities, where BIOES(U) stands for **B**egin, **I**nside, **O**utside, **E**nd and **S**ingle (**U**nique). BIOES has shown considerable performance improvements over BIO [35]. Entity type is simply appended to the BIO tags with a “-” in between, e.g. B-PER, I-PER, B-ORG, etc. Figure 1.2 shows an example represented with BIO and BIOES schemas.

A sequence classifier such as a CRF or an RNN is trained to label the tokens in a text with BIO tags.

Words	**BIO**	**BIOES**
Barack	B-PER	B-PER
Hussein	I-PER	I-PER
Obama	I-PER	I-PER
II	I-PER	E-PER
is	O	O
an	O	O
American	B-GPE	S-GPE
attorney	O	O
and	O	O
politician	O	O
who	O	O
served	O	O
as	O	O
the	O	O
44th	O	O
President	O	O
of	O	O
the	O	O
United	B-GPE	B-GPE
States	I-GPE	E-GPE
.	O	O

Table 1.2: Named entity tagging as a sequence model, showing BIO and BIOES schemas.

1.4.2 Evaluation

At the Fourth Message Understanding Conference (MUC-4) in 1992, F-measure was first introduced to measure the performance of the message understanding systems. F-measure provides a way of combining recall and precision to get a single measure which fall between recall and precision. Recall and precision can have relative weights in the calculation of the F-measure giving it the flexibility to be used for different applications [36]. The formula for calculating the F-measure is

$$F = \frac{(\beta^2 + 1.0) \cdot P \cdot R}{\beta^2 \cdot P + R}, \tag{1.1}$$

where P is precision, R is recall, and β is the relative importance given to recall over precision. In name tagging evaluation, β is set to 1.0 so that recall is as equally important as precision. It yields the standard F-1 metric for name tagging:

$$F = 2 \cdot \frac{P \cdot R}{P + R}, \tag{1.2}$$

In the F-1 metric, precision is the fraction of correctly tagged name spans among all the tagged named spans, while recall is the fraction of named spans that have been correctly tagged over the total amount of ground-truth named spans. More specifically, precision is computed as:

$$P = \frac{|\{\text{ground-truth names}\} \cap \{\text{tagged names}\}|}{|\{\text{tagged names}\}|} \tag{1.3}$$

, and recall is computed as:

$$R = \frac{|\{\text{ground-truth names}\} \cap \{\text{tagged names}\}|}{|\{\text{ground-truth names}\}|} \tag{1.4}$$

1.5 Contributions of the Thesis

The novel contributions of the thesis include the following aspects:

- To the best of our knowledge, we are the first to explore non-traditional linguistic resources, such as World Atlas of Linguistic Structure (WALS) and Central Intelligence Agency (CIA) Names, and proved that they are beneficial for deep neural networks based name tagging approaches. It illuminates the path of exploiting universal linguistic resources to improve multilingual low-resource language name tagging, besides the current research focus on machine learning model driven approaches.
- To overcome the ambiguity and insufficiency of local context in a sentence, we propose to use multiple levels of contextual information (local, document-level, and corpus-level) to improve name tagging performance. We designed a new global attention framework to exploit extra contextual information and achieve the state-of-the-art. Not limited to name tagging, the proposed idea can also be adopted to various tasks where global contextual information is important, such as relation extraction and event extraction.
- We introduce a new approach to train cross-lingual language model, and are the first to incorporate cross-lingual word embeddings and cross-lingual language model to name tagging for low-resource languages.

1.6 Related Publications

Some of the proposed research work has been published in the following peer-reviewed top NLP conferences.

- Name Tagging for Low-resource Incident Languages based on Expectation-driven Learning [37], *Boliang Zhang, Xiaoman Pan, Tianlu Wang, Ashish Vaswani, Heng Ji, Kevin Knight, Daniel Marcu, Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*

- Embracing Non-Traditional Linguistic Resources for Low-resource Language Name Tagging [38], *Boliang Zhang, Di Lu, Xiaoman Pan, Ying Lin, Halidanmu Abudukelimu, Heng Ji, Kevin Knight, Proceedings of the Eighth International Joint Conference on Natural Language Processing, 2017*

- Global Attention for Name Tagging, *Boliang Zhang, Spencer Whitehead, Lifu Huang, Heng Ji, Proceedings of the 22nd Conference on Computational Natural Language Learning, 2018*

- ELISA-EDL: A Cross-lingual Entity Extraction, Linking and Localization System [39], *Boliang Zhang, Ying Lin, Xiaoman Pan, Di Lu, Jonathan May, Kevin Knight, Heng Ji, Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations.*

Chapter 2
Related Work

2.1 Traditional Name Tagging

Name tagging is an active area of research for past three decades. A lot of progress has been made in detecting named entities, but name tagging still remains a major challenge.

Rule-based systems [40, 41] typically use hand-crafted rules and exhaustive lexicons or dictionaries to identify and classify named entities. They do not require annotated training data but highly rely on domain specific knowledge. Generally they achieve high precision because of the lexicons, while they suffer from low recall due to domain and language-specific rules and incomplete dictionaries. Another drawback of rule-based systems is the need of domain experts for constructing and maintaining the knowledge resources [42].

Statistical machine learning approaches soon take over rule-based systems in most scenarios as their better generalization and adaptation capability. They are probabilistic and use statistical models rather than deterministic rules. They learn to make predictions by training on example input and their expected output. Early machine learning models include Hidden Markov Model (HMM) [43, 44, 45, 46], Support Vector Machines (SVMs) [47], Conditional Random Fields (CRFs) [48, 49], and decision trees [50]. [45] first introduces HMM based name tagger on MUC-6 and MUC-7 data, achieving 96.6% and 94.1% F-1 score respectively. [51] compares the HMM with a SVM model on the CoNLL 2003 dataset. They proposed multiple new features, including multiple window sizes and orthographic features from neighboring words. They weight neighboring words features based on their position and class to balance positive and negative classes. On the English CoNLL 2003 data, they achieved 88.3% F-1 score [42]. [52] and [53] use a CRF model to capture the inter dependency between labels and achieved state-of-the-art results in the DrugNER task. Besides orthographic features, more features are used in CRF, including lexicon resources, word embeddings, dictionaries, etc.

The aforementioned statistical machine learning frameworks rely on domain-specific features. Table 2.1 presents a list of typical features for name tagging.

Table 2.2 presents some features for a sample English sentence “Barack Hussein Obama II is an American attorney and politician who served as the 44th President of the United

Features	**Descriptions**
Form	Lowercase form of (w_{-1}, w_0, w_{+1})
Case	whether w_0 is uppercase
Syllable	The first and last character of w_0
Affix	Affixes of (w_{-1}, w_0, w_{+1})
Gazetteer	whether w_0 is in gazetteers of PER, LOC, GPE, ORG, etc.
Embeddings	word embeddings learned from monolingual corpus

Table 2.1: Typical features for a feature-based name tagging system.

States.”.

Words	**Form**			**Case**	**Syllable**		**Gazetteer**			
	previous	word	next		first	last	PER	LOC	ORG	GPE
Barack	<SOS>	barack	hussein	1	B	k	1	0	0	0
Hussein	barack	hussein	obama	1	H	n	1	0	0	0
Obama	hussein	obama	ii	1	O	a	1	0	0	0
II	obama	ii	is	1	I	I	0	0	0	0
is	ii	is	an	0	i	s	0	0	0	0
an	is	an	american	0	a	n	0	0	0	0
American	an	american	attorney	1	A	n	0	0	1	0
attorney	american	attorney	and	0	a	y	0	0	0	0
and	attorney	and	politician	0	a	d	0	0	0	0
politician	and	politician	who	0	p	n	0	0	0	0
who	politician	who	served	0	w	o	0	0	0	0
served	who	served	as	0	s	d	0	0	0	0
as	served	as	the	0	a	s	0	0	0	0
the	as	the	44th	0	t	e	0	0	0	0
44th	the	44th	President	0	4	h	0	0	0	0
President	44th	President	of	1	P	t	1	0	0	0
of	President	of	the	0	o	f	0	0	0	0
the	of	the	United	0	t	e	0	0	0	0
United	the	United	States	1	U	d	0	0	0	1
States	United	States	.	1	S	s	0	0	0	1
.	States	.	<EOS>	0	.	.	0	0	0	0

Table 2.2: An example of name tagging features for an English sentence.

The effectiveness of features highly relies on the application, genre, media, and language. For example, morphological features, such as affixes and suffixes, are critical for morphology rich languages but of little use with languages that do not have morphology, such as Chinese. Compared to rule-based systems, feature-based machine learning methods have improved generalization and adaptation abilities, but they merely turn human curated rules to "feature engineering" which still depends on hand-crafted and domain specific features.

2.2 Neural Name Tagging

[54] is the first work to use neural network architecture for name tagging. They construct feature vectors from orthographic features, in the same way as feature-based machine learning method, but use multi-layer feed forward neural networks for label prediction. Later work replaced these manually constructed feature vectors with word embeddings [55, 42], which are representations of words in n-dimensional space, typically learned over large collections of unlabeled monolingual corpus through an unsupervised process such as continuous bag-of-word (CBOW) and skip-gram model in [22, 42]. Afterwards, because of the big advantages on domain adaptation and generalization, neural network based approaches became dominant in the field of name tagging. Approaches proposed in this thesis are extensions of the basic neural networks towards multilingual name tagging.

The input of most neural name tagging models include word representations which are embedding vectors that are randomly initialized or pre-trained from large monolingual corpora, character representations which are computed by Recurrent Neural Networks (RNN) or Convolutional Neural Networks (CNN) over characters of each word, and optionally, a set of linguistic features that are crafted from external resources.

Word level architectures - In this architecture, the words of a sentence are given as input to a Recurrent Neural Networks (RNN) and each word is represented by its word embedding [42].

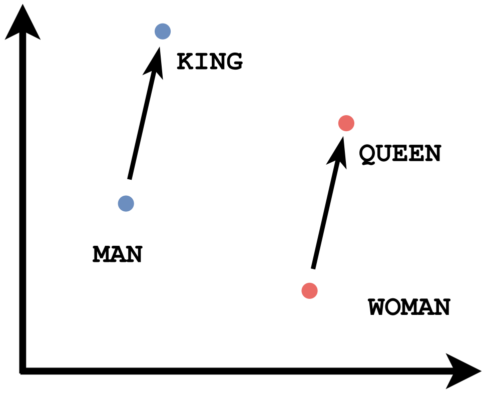


Figure 2.1: A toy word embedding example.

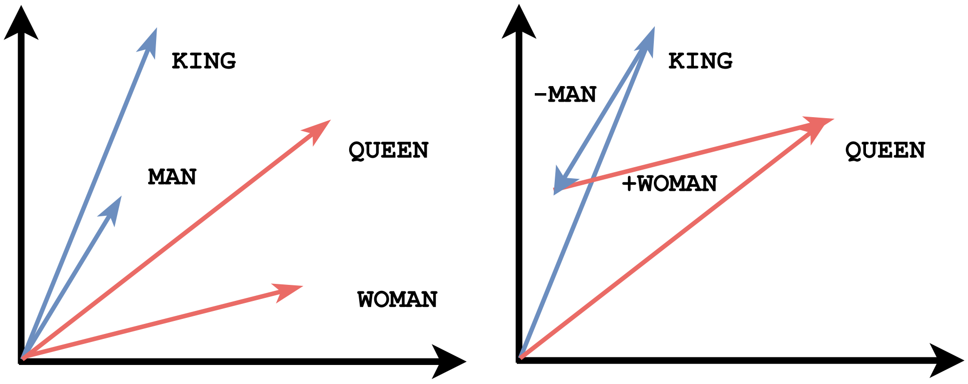


Figure 2.2: Explanation of word embedding composition: $w_{KING} - w_{MAN} + w_{WOMAN} = w_{QUEEN}$.

Word embeddings are essential components of neural networks. They are vectors of real numbers in a high-dimensional space, and conceptually, they are distributional word semantic representations that aim to quantify and categorize semantic similarities between words. Word embeddings are unsupervisedly trained from monolingual corpus via various methods such as Continuous Bag-of-words (BOW) and skip gram [22]. Figure 2.1 is a toy word embedding example that shows vector offsets for three word pairs illustrating the gender relation [56]. Figure 2.2 shows a famous analogy of relations between different word embeddings: $w_{KING} - w_{MAN} + w_{WOMAN} = w_{QUEEN}$. The theory behind this is beyond the scope of this chapter, we refer the reader to [56].

Pre-trained from a large monolingual corpus, word embeddings are features that capture each word's global semantic representation in the entire corpus. Predicting the tag for each token needs evidence from both of its previous context and future context in the entire sentence. Given a specific context, we apply Recurrent Neural Networks on word embeddings to retrieve context specific features. Bi-directional Long Short-term Memory (Bi-LSTM) networks [57] processes each sequence in both directions with two separate hidden layers, which are then fed into the same output layer. Also, there are strong classification dependencies among name tags in a sequence. For example, "I-LOC" cannot follow "B-ORG". Conditional Random Fields (CRFs) model, which is particularly good at jointly modeling tagging decisions, can be built on top of the Bi-LSTM networks. Figure 2.3 shows a word level name tagging neural architecture using Bi-LSTM and CRF.

Huang et al. [58] and Lample et al. [59] propose the first neural architecture consisting

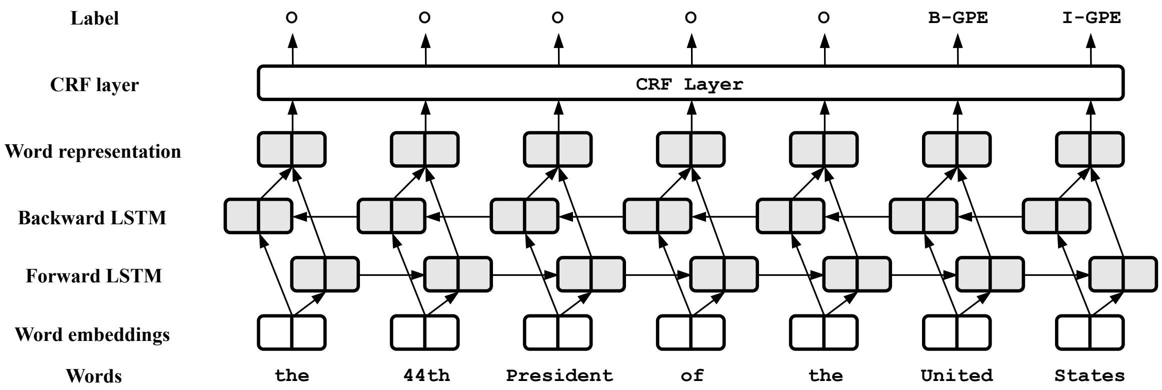


Figure 2.3: Word level Bidirectional LSTM name tagging architecture.

of a Bi-LSTM encoder and CRF output layer (Bi-LSTM CRF). This architecture has been widely explored and demonstrated to be effective for sequence labeling tasks.

Character level architectures - As characters are the minimum unit of a sentence, architectures combining word context and the characters of a word have been proven to be strong name tagging approaches that need little domain-specific knowledge or resources. We apply a Bi-LSTM or Convolutional Neural Networks (CNN) [60] on the characters of each word in each sentence to generate a sequence of character level word representations. Then each word is represented as a combination of a word embedding and a Bi-LSTM or CNN over the characters of the word, followed with a Bi-LSTM layer over the word representations of a sentence. Figure 2.4 presents a word+character level embedding architecture. Efforts incorporating character level compositional word embeddings into the Bi-LSTM CRF architecture has improved performance include [60, 61, 62, 30, 29].

2.3 Multilingual and Low-resource Language Name Tagging

Recently, multilingual and low-resource language name tagging has drawn attention from the NLP community as the neural networks learn features automatically from training data and do not require language specific knowledge, which makes it possible to adapt neural architectures that highly perform on one language to other languages. [63, 59] achieve state-of-the-art scores using the same Bi-LSTM+Character embeddings architectures on CoNLL 2003 English, Dutch, German and Spanish dataset. Other neural multilingual name tagging efforts are as follows, [64, 65, 66, 67] use bilingual labeled data, [68] uses naturally partially annotated data such as Wikipedia, [69, 49, 70, 71, 41, 72] that uses traditional non-neural

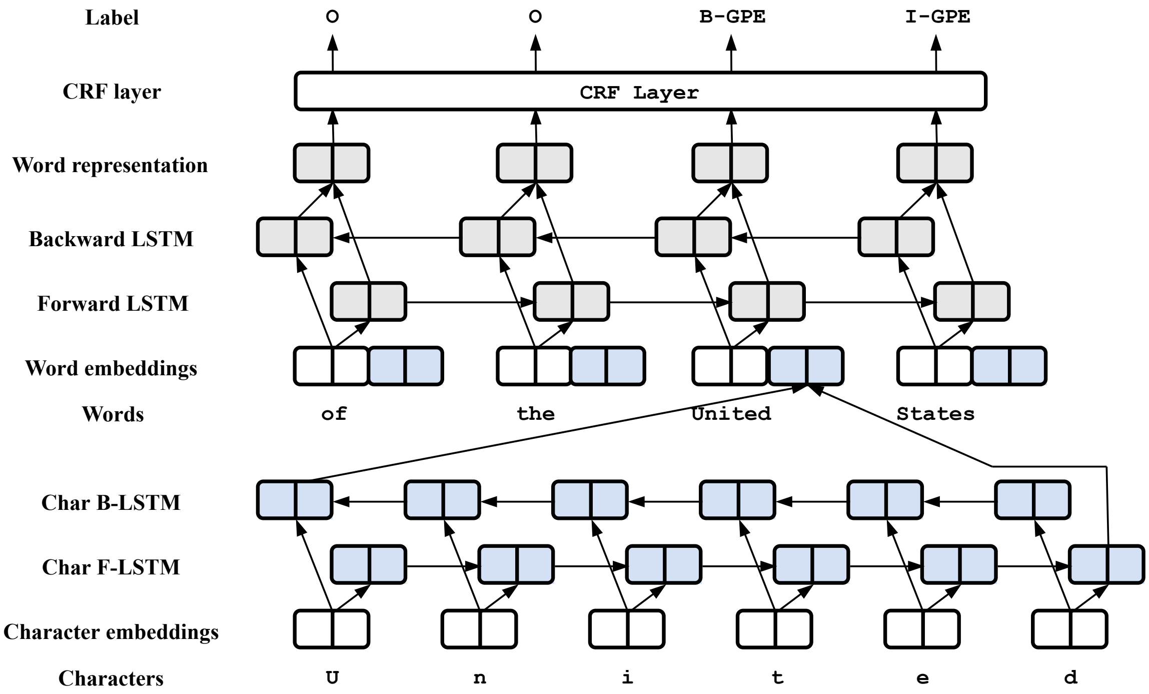


Figure 2.4: Word+character level Bidirectional LSTM name tagging architecture.

unsupervised learning approaches.

Most of the aforementioned methods require labeled data. In situations where massive clean annotations are insufficient or unavailable, the supervised learning methods including DNN, are sensitive to noise, and are unable to achieve satisfying performance.

Various automatic annotation generation methods have been proposed to compensate the data requirement, including knowledge base driven distant supervision [73, 74, 75], cross-lingual projection [76, 65, 66, 67, 77, 78], and exploiting naturally existing noisy annotations such as Wikipedia markups [79, 80, 81, 82, 83, 84, 85]. [63, 86] apply various transferring methods to transfer name tagging capability from high-resource languages to low resource languages. However, they still heavily rely on information redundancy, and they are sensitive to noise.

Chapter 3
Noisy Annotation Acquisition

3.1 “Chinese Room”

In many emergent situations such as disease outbreaks and natural disasters, there is great demand to rapidly develop a Natural Language Processing (NLP) system, such as name tagger, for a “surprise” Incident Language (IL) with very few resources. Traditional supervised learning methods that rely on large-scale manual annotations would be too costly.

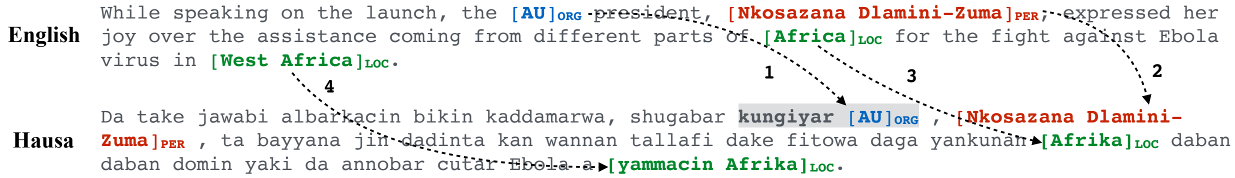
We designed an interface where test sentences are presented to the player one by one. When the player clicks token, the interface will display up to 100 manually labeled Tibetan sentences that include this token. The player can also see translations of some common words and a small gazetteer of common names (800 entries) in the interface.

14 players who don’t know Tibetan joined the game. Their name tagging F-scores ranged from 0% to 94%. We found that good players usually bring in some kind of “***expectations***” derived from their own native languages, or general linguistic knowledge, or background knowledge about the scenario. Then they actively search, confirm, adjust and update these expectations during tagging. For example, they know from English that location names are often ended with suffix words such as “*city*” and “*country*”, so they search for phrases starting or ending with the translations of these suffix words. After they successfully tag some seeds, they will continue to discover more names based on more expectations. For example, if they already tagged an organization name A, and now observe a sequence matching a common English pattern “*[A (Organization)]’s [Title] [B (Person)]*”, they will tag B as a person name. And if they know the scenario is about Ebola, they will be looking for a phrase with translation similar to “*West Africa*” and tag it as a location. Similarly, based on the knowledge that names appear in a conjunction structure often have the same type, they propagate high-confidence types across multiple names. They also keep gathering and synthesizing common contextual patterns and rules (such as position, frequency and length information) about names and non-names to expand their expectations. For example, after observing a token frequently appearing between a subsidiary and a parent organization, they will predict it as a preposition similar to “*of*” in English, and tag the entire string as a nested organization. Based on these lessons learned from this game, we develop an

annotation platform to allow non-native speakers to annotate names in LL text.

3.2 Cross-lingual Name Projection

In low-resource settings where few clean annotations are available, we could try to automatically generate some annotations to train the supervised model. For instance, we can project automatic annotations from a HL to a LL through parallel data. Figure 3.1 shows an example of projecting English automatic name annotations to Hausa through a parallel sentence pair.



While speaking on the launch, the [AU]$_{ORG}$ president, [Nkosazana Dlamini-Zuma]$_{PER}$, expressed her joy over the assistance coming from different parts of [Africa]$_{LOC}$ for the fight against Ebola virus in [West Africa]$_{LOC}$.

Da take jawabi albarkacin bikin kaddamarwa, shugabar kungiyar [AU]$_{ORG}$, [Nkosazana Dlamini-Zuma]$_{PER}$, ta bayyana jin dadinta kan wannan tallafi dake fitowa daga yankunan [Afrika]$_{LOC}$ daban daban domin yaki da annobar cutar Ebola a [yammacin Afrika]$_{LOC}$.

* Projection 1 is incorrect and results in a noisy instance in the automatically generated Hausa annotations. The correct name mention is “kungiyar AU (Africa Union)” instead of “AU”.

Figure 3.1: Noisy Training Data Generation by Projecting English Automatic Name Annotations to Hausa.

We use S to denote the sentences in LL and T to denote the sentences in HL. We apply Stanford English name tagger [87] on T and project English names onto S, using the following measurements to determine whether a candidate LL name string n_l matches an expected English name n_e: (1) If the edit distance between n_e and n_l is not greater than two. (2) We check the pronunciations of n_e and n_l based on Soundex [88], Metaphone [89] and NYSIIS [90] algorithms. We consider two codes match if their edit distance is not greater than two. (3) If n_e and n_l are aligned in the parallel data by running GIZA++ word alignment tool [91]. In this way we obtain an automatically generated noisy training data set.

3.3 Wikipedia Knowledge Base (KB) mining

In addition to unstructured documents, we also try to leverage structured English knowledge bases (KBs) such as DBpedia.[2] Each entry is associated with a set of types such as `Company`, `Actor` and `Agent`. We utilize the Abstract Meaning Representation corpus [92]

[2] http://dbpedia.org

which contains both entity type and linked KB title annotations, to automatically map $9,514$ entity types in DBPedia to three main entity types of interest: Person (PER), Location (LOC) and Organization (ORG).

We need to select sentences for inclusion in our training corpus for which we are confident of having correctly labeled all named entities. We use various criteria to filter out unreliable sentences including: (1) sentences that shorter than five tokens or longer than a hundred tokens are ignored, (2) sentences that are all capitalized are ignored, and (3) sentences that have more than fifty percent tokens labeled as named entities are ignored.

Chapter 4
Incorporating Linguistic resources: Expectation-driven Learning

4.1 Time Zero: Language Universals

At time zero, we aim to rapidly build a rule-based name tagging system by language universal resources. This system is used as a baseline approach. First we use some language universal rules, gazetteers and patterns to generate a binary feature vector $F = \{f_1, f_2, ...\}$ for each token. Table 4.1 shows these features along with examples. An identification rule is $r_I = \langle T_I, f = \{f_a, f_b, ...\}\rangle$ where T_I is a "B/I/O" tag to indicate the beginning, inside or outside of a name, and $\{f_a, f_b, ...\}$ is a set of selected features. If the features are all matched, the token will be tagged as T_I. Similarly, a classification rule is $r_C = \langle T_C, f = \{f_a, f_b, ...\}\rangle$, where T_C is "Person/Organization/Location". These rules are triggered in order, and some examples are as follows: ⟨B, {AllUppercased}⟩, ⟨PER, {PersonGaz}⟩, ⟨ORG, {Capitalized, LongLength}⟩, etc.

Features	Examples (Feature name is underlined)
in English Gazetteer	- **PerGaz**: person $(472, 765)$; **LocGaz**: location $(211, 872)$; **OrgGaz**: organization $(124, 403)$; **Title** (889); **NoneName** $(2, 380)$.
Case	- **Capitalized**; - **AllUppercased**; - **MixedCase**
Punctuation	- **IternalPeriod**: includes an internal period
Digit	- **Digits**: consisted of digits
Length	- **LongLength**: a name including more than 4 tokens is likely to be an ORG
TF-IDF	- **TF-IDF**: if a capitalized word appears at the beginning of a sentence, and has a low TF-IDF, then it's unlikely to be a name
Patterns	- **Pattern1**: "*Title* ⟨ PER Name ⟩"
	- **Pattern2**: "$\langle PERName\rangle, 00*,$" where 00 are two digits
	- **Pattern3**: "$[\langle Name_i\rangle ...], \langle Name_n - 1\rangle\langle singleterm\rangle\langle Name_n\rangle$" where all names have the same type.
Multi-occurrences	- **MultipleOccurrence**: If a word appears in both uppercased and lowercased forms in a single document, it's unlikely to be a name.

Table 4.1: Universal Name Tagger Features

4.2 Expectation Learning

4.2.1 Approach Overview

Figure 4.1 illustrates our overall approach of acquiring various expectations, by simulating the strategies human players adopted during the Tibetan Room game. Next we will present details about discovering expectations from each source.

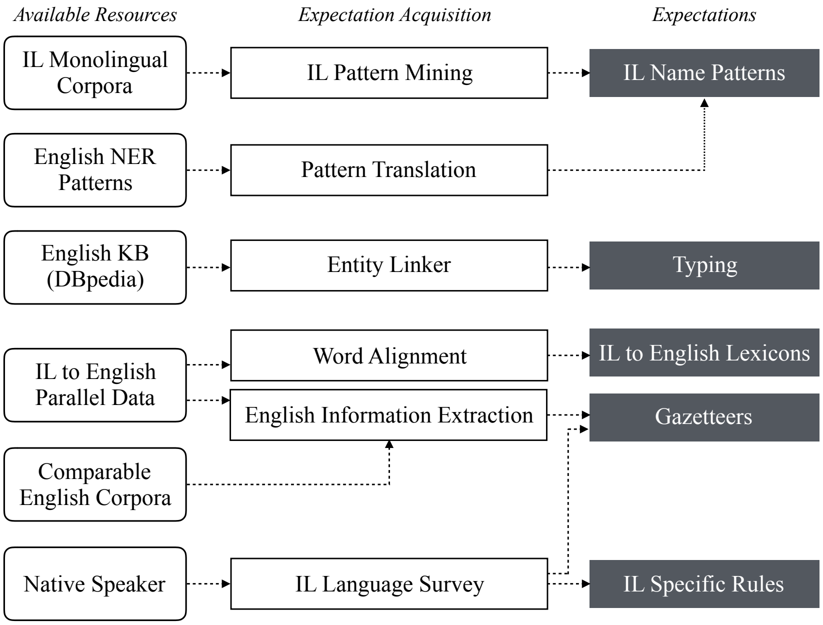


Figure 4.1: Expectation Driven Name Tagger Overview

4.2.2 Survey with Native Speaker

The best way to understand a language is to consult people who speak it. We introduce a human-in-the-loop process to acquire knowledge from native speakers. To meet the needs in the emergent setting, we design a comprehensive survey that aims to acquire a wide-range of IL-specific knowledge from native speakers in an efficient way. The survey categorizes questions and organizes them into a tree structure, so that the order of questions is chosen based on the answers of previous questions. The survey answers are then automatically translated into rules, patterns or gazetteers in the tagger. Some example questions are shown in Table 4.2.

True/False Questions
1. The letters of this language have upper and lower cases
2. The names of people, organizations and locations start with a capitalized (uppercased) letter
3. The first word of a sentence starts with a capitalized (uppercased) letter
4. Some periods indicate name abbreviations, e.g., St. = Saint, I.B.M. = International Business Machines.
5. Locations usually include designators, e.g., in a format like "country United states", "city Washington"
6. Some prepositions are part of names

Text input
1. Morphology: please enter preposition suffixes as many as you can (e.g. "'da" in "Ankara'da yaşıyorum (I live in Ankara)" is a preposition suffix which means "in").

Translation
1. Please translate the following English words and phrases:
- organization suffix: agency, group, council, party, school, hospital, company, office, ...
- time expression: January, ..., December; Monday, ..., Sunday; ...

Table 4.2: Survey Question Examples

4.2.3 Mono-lingual Expectation Mining

We use a bootstrapping method to acquire IL patterns from unlabeled mono-lingual IL documents. Following the same idea in [93, 94], we first use names identified by high-confident rules as seeds, and generalize patterns from the contexts of these seeds. Then we evaluate the patterns and apply high-quality ones to find more names as new seeds. This process is repeated iteratively.[3]

We define a pattern as a triple $\langle left, name, right \rangle$, where *name* is a name, left and right[4] are context vectors with weighted terms (the weight is computed based on each token's tf-idf score). For example, from a Hausa sentence "***gwamnatin kasar Sin ta*** *samar wa kasashen yammacin Afirka ... (the Government of China has given ... products to the West African countries)*", we can discover a pattern:

- $left$: ⟨**gwamnatin** (goevernment), 0.5⟩, ⟨**kasar** (country), 0.6⟩
- *name*: ⟨**Sin** (China), 0.5⟩
- *right*: ⟨**ta** (by), 0.2⟩

[3] We empirically set the number of iterations as 2 in this paper.

[4] $left$ and *right* are the context three tokens before and after the name

This pattern matches strings like "*gwamnatin kasar Fiji ta (by the government of Fiji)*".

For any two triples $t_i = \langle l_i, name_i, r_i \rangle$ and $t_j = \langle l_j, name_j, r_j \rangle$, we comput e their similarity by:

$$Sim(t_i, t_j) = l_i \cdot l_j + r_i \cdot r_j$$

We use this similarity measurement to cluster all triples and select the centroid triples in each cluster as candidate patterns.

Similar to [93], we evaluate the quality of a candidate pattern P by:

$$Conf(P) = \frac{P_{positive}}{(P_{positive} + P_{negative})}$$

,where $P_{positive}$ is the number of positive matches for P and $P_{negative}$ is the number of negative matches. Due to the lack of syntactic and semantic resources to refine these lexical patterns, we set a conservative confidence threshold 0.9.

4.2.4 Cross-lingual Expectation Projection

Name tagging research has been done for high-resource languages such as English for over twenty years, so we have learned a lot about them. We collected 1,362 patterns from English name tagging literature [95, 96, 97]. Some examples are listed below:

- $\langle \{\}, \{PER\}, \{< say >, < . >\} \rangle$
- $\langle \{< headquarter >, < in >\}, \{LOC\}, \{\} \rangle$
- $\langle \{< secretary >, < of >\}, \{ORG\}, \{\} \rangle$
- $\langle \{< in >, < the >\}, \{LOC\}, \{< area >\} \rangle$

4.3 Supervised Active Learning

We anticipated that not all expectations can be encoded as explicit rules and patterns, or covered by projected names, therefore for comparison we introduce a supervised method with pool-based active learning to learn implicit expectations (features, new names, etc.) directly from human data annotation. We exploited basic lexical features including ngrams, adjacent tokens, casing information, punctuations and frequency to train a Conditional Random Fields (CRFs) [98] based model through active learning [99].

We segment documents into sentences and use each sentence as a training unit. Let $\mathbf{x}_b^*$ be the most informative instance according to a query strategy $\phi(\mathbf{x})$, which is a function used to evaluate each instance $\mathbf{x}$ in the unlabeled pool U. Algorithm 1 illustrates the procedure.

Algorithm 1 Pool-based Active Learning

```
$L \leftarrow$ labeled set, $U \leftarrow$ unlabeled pool
$\phi(\cdot) \leftarrow$ query strategy, $B \leftarrow$ query batch size
$M \leftarrow$ maximum number of tokens
while Length($L$)$< M$ do
    $\theta = \text{train}(L)$;
    for $b \in \{1, 2, ..., B\}$ do
        $\mathbf{x}_b^* = \arg\max_{x \in U} \phi(\mathbf{x})$
        $L = L \cup \{\mathbf{x}_b^*, \text{label}(\mathbf{x}_b^*)\}$
        $U = U - \mathbf{x}_b^*$
    end for
end while
```

[100] proposed an entropy measure for active learning for image retrieval task. We compared it with other measures proposed by [101] and found that **sequence entropy (SE)** is most effective for our name tagging task. We use ϕ^{SE} to represent how informative a sentence is:

$$\phi^{SE}(\mathbf{x}) = -\sum_{t=1}^{T}\sum_{m=1}^{M} P_\theta(y_t = m) log P_\theta(y_t = m)$$

, where T is the length of $\mathbf{x}$, m ranges over all possible token labels and $P_\theta(y_t = m)$ is the probability when y_t is tagged as m.

4.4 Cost-aware Combination

A new requirement for IL name tagging is a **Linguistic Workflow Generator**, which can generate an activity schedule to organize and maximize the use of acquired expectations to yield optimal F-scores within given time bounds. Therefore, the input to the IL name tagger is not only the test data, but also a time bound for development (1 hour, 2 hours, 24 hours, 1 week, 1 month, etc.).

Figure 4.2 illustrates our cost-aware expectation composition approach. Given some IL documents as input, as the clock ticks, the system delivers name tagging results at time 0 (immediately), time 1 (e.g., in one hour) and time 2 (e.g., in two hours). At time 0, name tagging results are provided by the universal tagger described in Section 4.1. During the first hour, we can either ask the native speaker to annotate a small amount of data for supervised active learning of a CRFs model, or fill in the survey to build a rule-based tagger.

Language	IL Test Docs	Name	Unique Name	IL Dev. Docs	IL-English Docs
Bengali	100	4,713	2,820	12,495	169
Hausa	100	1,619	950	13,652	645
Tagalog	100	6,119	3,375	1,616	145
Tamil	100	4120	2,871	4,597	166
Thai	100	4,954	3,314	10,000	191
Turkish	100	2,694	1,323	10,000	484
Yoruba	100	3,745	2,337	427	252

Table 4.3: Data Statistics

We estimate the confidence value of each expectation-driven rule based on its precision score on a small development set of ten documents. Then we apply these rules in the priority order of their confidence values. When the results of two taggers are conflicting on either mention boundary or type, if the applied rule has high confidence we will trust its output, otherwise adopt the CRFs model's output.

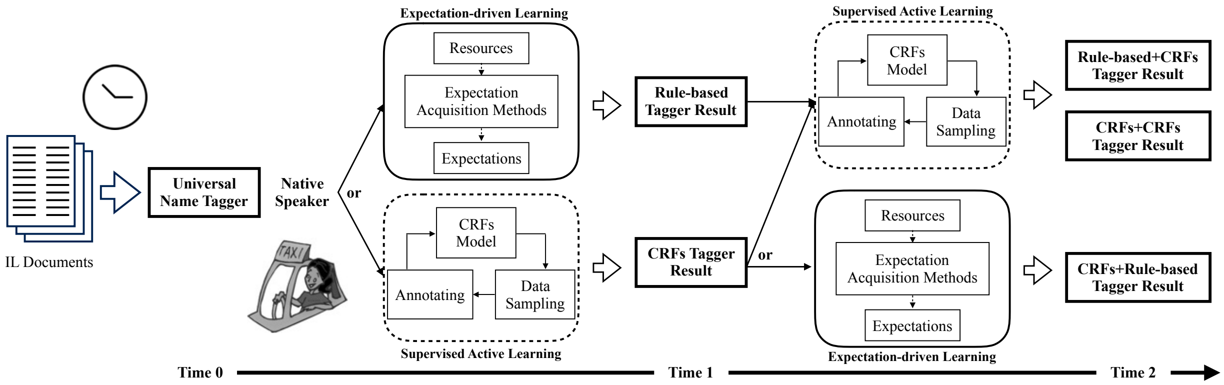


Figure 4.2: Cost-aware Expectation Composition

4.5 Experiments

In this section we will present our experimental details, results and observations.

4.5.1 Data

We evaluate our framework on seven low-resource incident languages: Bengali, Hausa, Tagalog, Tamil, Thai, Turkish and Yoruba, using the ground-truth name tagging annotations from the DARPA LORELEI program.[5] Table 4.3 shows data statistics.

[5] http://www.darpa.mil/program/low-resource-languages-for-emergent-incidents

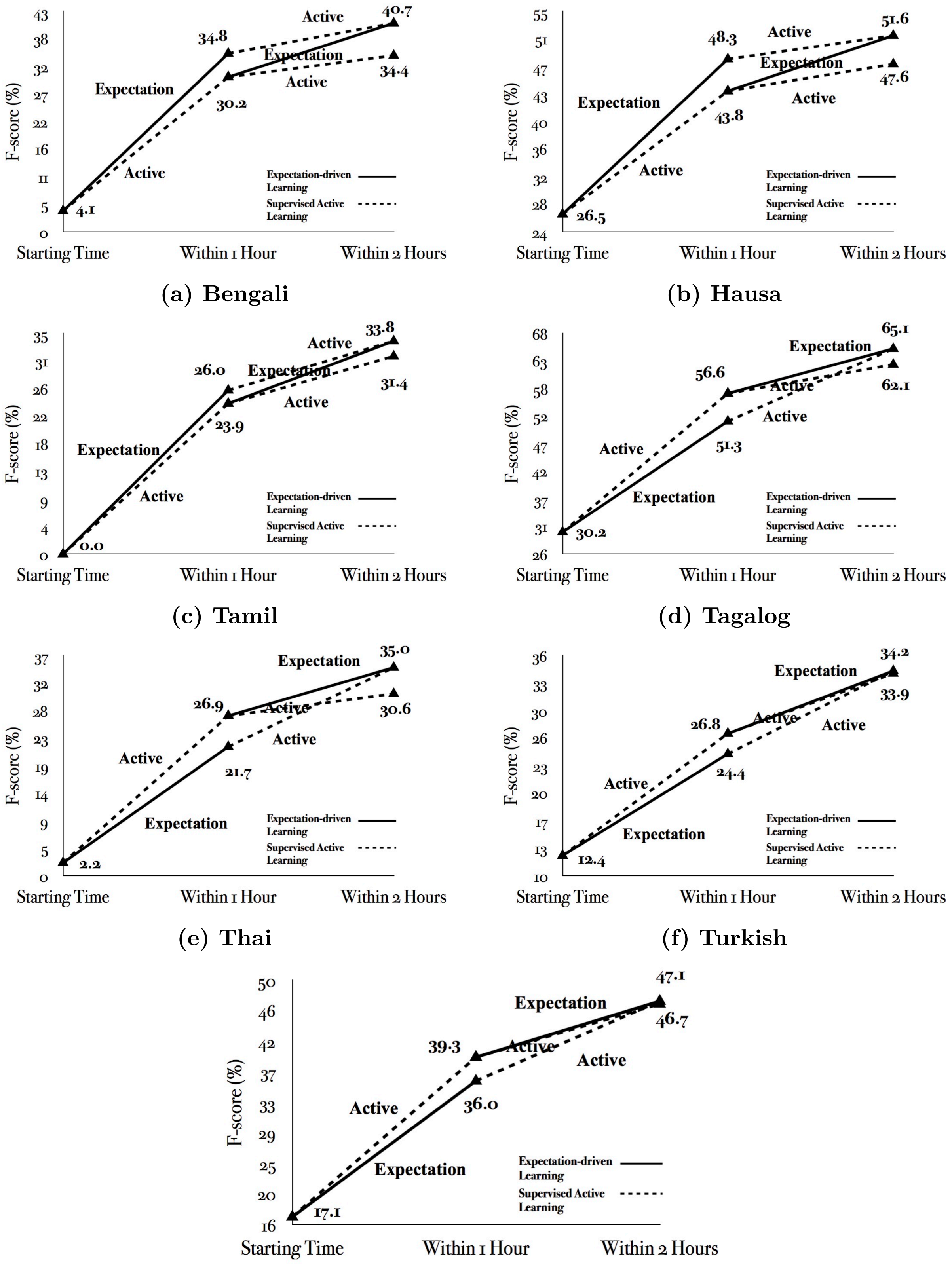
4.5.2 Cost-aware Overall Performance

We test with three checking points: starting time, within one hour, and within two hours. Based on the combination approach described in Section 4.4, we can have three possible combinations of the expectation-driven learning and supervised active learning methods during two hours: (1) expectation-driven learning + supervised active learning; (2) supervised active learning + expectation-driven learning; and (3) supervised active learning for two hours. Figure 4.3 compares the overall performance of these combinations for each language.

We can see that our approach is able to rapidly set up a name tagger for an IL and achieves promising performance. During the first hour, there is no clear winner between expectation-driven learning or supervised active learning. But it's clear that supervised active learning for two hours is generally not the optimal solution. Using Hausa as a case study, we take a closer look at the supervised active learning curve as shown in Figure 4.4. We can see that supervised active learning based on simple lexical features tends to converge quickly. As time goes by it will reach its own upper-bound of learning and generalizing linguistic features. In these cases our proposed expectation-driven learning method can compensate by providing more explicit and deeper IL-specific linguistic knowledge.

4.5.3 Comparison of Expectation Discovery Methods

Table 4.4 shows the performance gain of each type of expectation acquisition method. IL gazetteers covered some common names, especially when the universal case-based rules failed at identifying names from non-Latin languages. IL name patterns were mainly effective for classification. For example, the Tamil name "கத்தோலிக்கன் சிரியன் வங்கியில (Catholic Syrian Bank)" was classified as an organization because it ends with an organization suffix word "வங்கியில(bank)". The patterns projected from English were proven very effective at identifying name boundaries. For example, some non-names such as titles are also capitalized in Turkish, so simple case-based patterns produced many spurious names. But projected patterns can fix many of them. In the following Turkish sentence, "*Ancak Avrupa Birliği Dış İlişkiler Sorumlusu Catherine Ashton,...(But European Union foreign policy chief Catherine Ashton,...)*", among all these capitalized tokens, after we confirmed "*Avrupa Birliği (European Union)*" as an organization and "*Dış İlişkiler Sorumlusu (foreign policy chief)*" as a title, we applied a pattern projected from English "*[Organization] [Title] [Person]*" and



(a) Bengali

(b) Hausa

(c) Tamil

(d) Tagalog

(e) Thai

(f) Turkish

(g) Yoruba

Figure 4.3: Comparison of methods combining expectation-driven learning and supervised active learning given various time bounds

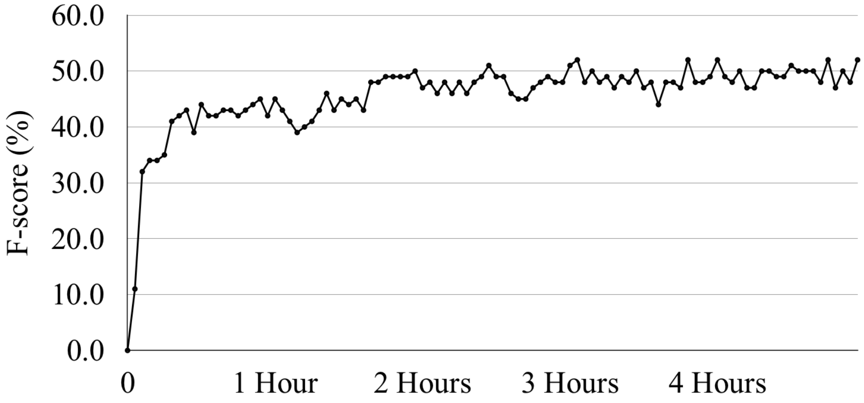


Figure 4.4: Hausa Supervised Active Learning Curve

Methods	Bengali	Hausa	Tamil	Tagalog	Thai	Turkish	Yoruba
Universal Rules	4.1	26.5	0.0	30.2	2.2	12.4	17.1
+IL Gazetteers	29.7	32.1	21.8	34.3	18.9	17.3	26.9
+IL Name Patterns	31.2	33.8	22.9	35.1	18.9	19.1	28.0
+IL to English Lexicons	31.3	35.2	24.0	38.0	20.5	19.6	29.4
+IL Survey with Native Speaker	34.1	40.6	25.6	45.9	21.6	39.3	30.2
+KB Linking based Typing	34.8	48.3	26.0	51.3	21.7	43.6	36.0

Table 4.4: Contributions of Various Expectation Discovery Methods (F-score %)

successfully identified "*Catherine Ashton*" as a person. Cross-lingual entity linking based typing successfully enhanced classification accuracy, especially for languages where names often appear the same as their English forms and so entity linking achieved high accuracy. For example, "*George Bush*" keeps the same in Hausa, Tagalog and Yoruba as English.

4.5.4 Impact of Supervised Active Learning

Figure 4.5 shows the comparison of supervised active learning and passive learning (random sampling in training data selection). We asked a native speaker to annotate Chinese news documents in one hour, and estimated the human annotation speed approximately as 7,000 tokens per hour. Therefore we set the number of tokens as 7,000 for one hour, and 14,000 for two hours. We can clearly see that supervised active learning significantly outperforms passive learning for all languages, especially for Tamil, Tagalog and Yoruba. Because of the rich morphology in Turkish, the gain of supervised active learning is relatively small because simple lexical features cannot capture name-specific characteristics regardless of the size of labeled data. For example, some prepositions (e.g., "*nin (in)*") can be part of

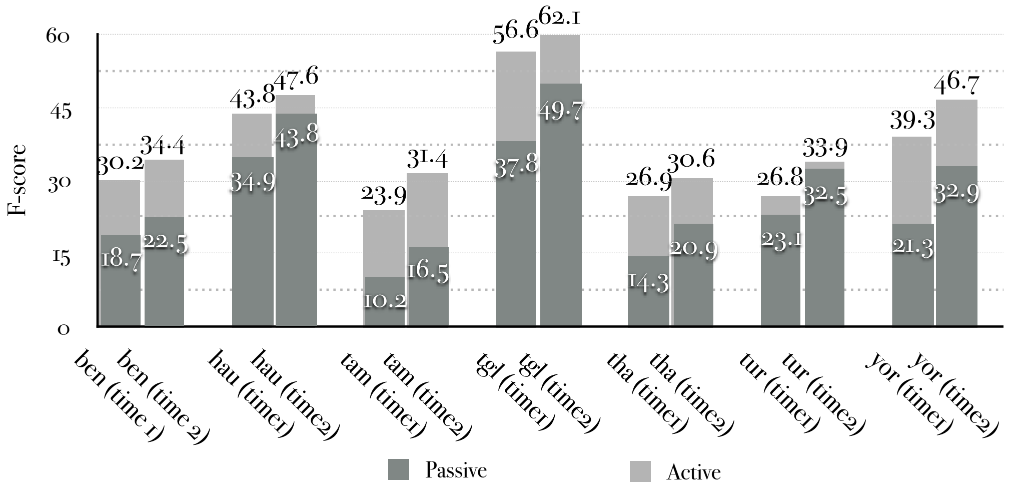


Figure 4.5: Active Learning vs. Passive Learning (%)

the names, so it's difficult to determine name boundaries, such as "*<ORG Ludian bölgesi* ***hastanesi>nin*** *(in <ORG Ludian Hospital>)*"

Chapter 5
Incorporating Linguistic resources: Non-traditional Linguistic Resources

5.1 Non-traditional Linguistic Resources

There is a general agreement that Deep Neural Networks provides a general, powerful underlying model for Information Extraction (IE), confirmed by improved state-of-the-art performance on many tasks such as name tagging [102, 103], relation classification [104, 105, 106, 107] and event detection [106, 108, 109, 110, 111]. For example, our experiments on several languages show that a DNN-based name tagger generally outperforms (up to 6% F-score gain) a Conditional Random Fields (CRFs) model trained from the same labeled data and feature set. DNN architecture is attractive to couple with character/word embeddings for IE tasks because it is easy to learn and usually effective enough to eliminate the need of explicit linguistic feature design.

In order to compensate the DNN name tagging data requirement, various automatic annotation generation methods have been proposed, including knowledge base driven distant supervision [73, 74, 75], cross-lingual projection [76, 65, 66, 67, 77, 78], and leveraging naturally existing noisy annotations such as Wikipedia markups [79, 80, 81, 82, 83, 84, 85]. Annotations produced from these methods are usually very noisy, while DNN is sensitive to noise just like many other machine learning methods. Our name tagging experiment shows that the F-score of the same DNN model learned from noisy training data is 20-30% lower than that trained from clean data. One major reason is that most of these methods solely rely on implicit embedding features in order to be (almost) language-independent.

On an almost parallel research avenue, linguists and domain experts have created a wide variety of multi-lingual resources, such as World Atlas of Linguistic Structure (WALS) [18], Central Intelligence Agency (CIA) Names, grammar books, and survival guides. Such resources have been largely ignored by the mainstream statistical NLP research, because they were not specifically designed for NLP purpose at the first place and they are often far from complete. Thus they are not immediately actionable - converted into features, rules or patterns for a target NLP application. In this chapter we design various methods to convert them into machine readable features for a new DNN architecture. Limited previous work only used them for resource building (e.g., [112]) or studying word order typology [113].

We aim to answer the following research questions: How to effectively acquire linguistic knowledge from non-traditional resources, and represent them for computational models? How much further gain can be obtained in addition to traditional resources?

5.2 Approach Overview

5.2.1 Baseline's Sensitiveness to Noise

In this section, we present the method to acquire various noise level training data, and show the sensitiveness of the DNN baseline to noise.

Given a parallel corpus of LL and HL (English), we use S to denote the sentences in LL and T to denote the sentences in HL. Utilizing the approach mentioned in Section 3.2, we obtain an automatically generated noisy training data set from parallel sentences. We denote $Train_{noise}$ as the obtained noisy training data, and $Train_{clean}$ as the ground truth which is manually created by human annotators on set S. We mix $Train_{noise}$ and $Train_{clean}$ in different proportions to obtain a training set $Train_{mix}$ on various noise levels. We define ***noise level*** as $1 - fscore(Train_{mix})$ where the f-score of $Train_{mix}$ is computed against $Train_{clean}$. For example, when $Train_{mix}$ is full of manually created clean data, the noise level is 0; when we mix half $Train_{noise}$ and half $Train_{clean}$ of the Hausa data, the f-score of $Train_{mix}$ is 80.1%, and the noise level is 19.9%.

To learn embeddings, we use 12,624 Hausa documents from the LORELEI program, and use 288,444 Turkish documents and 128,763 Uzbek documents from a June 2015 Wikipedia dump. Figure 5.1 shows the performance of the baseline tagger trained from $Train_{mix}$ for three languages. We can clearly see that the performance drops rapidly as the training data includes more noise.

5.2.2 A New Improved Model

We propose to acquire non-traditional linguistic resources and encode them as new actionable features (Section 5.3). In Figure 5.2, we design three integration methods to incorporate explicit linguistic features into Bi-LSTM networks: (1) concatenate the linguistic features and word embeddings at the input level, (2) concatenate the linguistic features and the bidirectional encodings of each token before feeding them into the output layer that

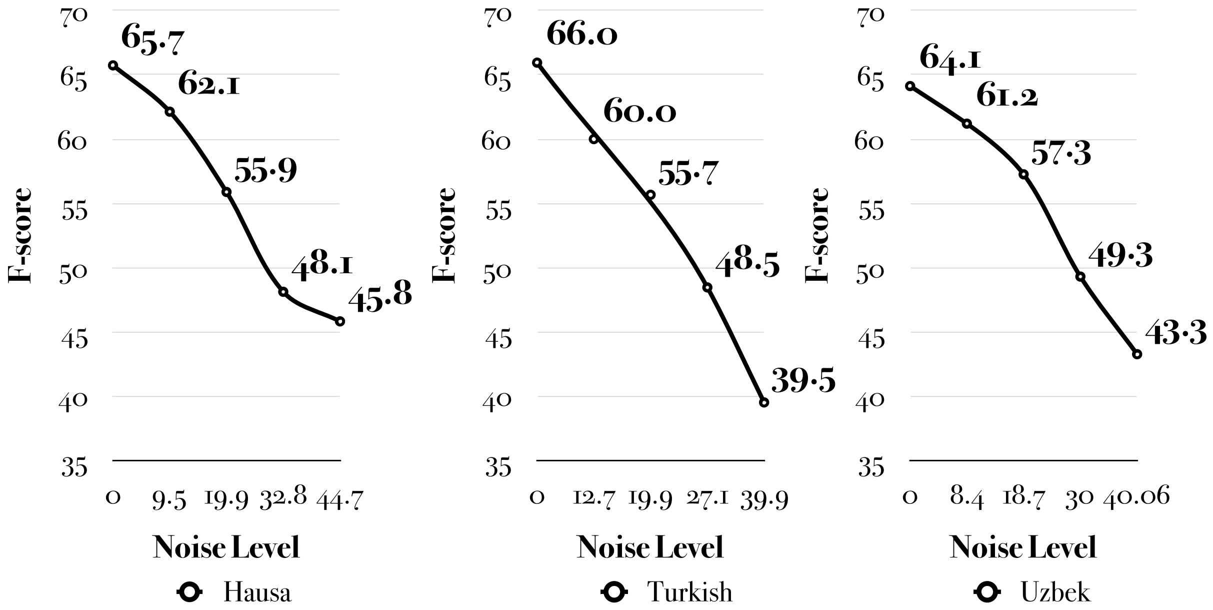


Figure 5.1: Performance of baseline DNN Name Taggers Trained from Data with Various Noise Levels (The noise level is created by assigning the proportion of $Train_{noise}$ in $Train_{mix}$ as 0%, 25%, 50%, 75% and 100% respectively.)

computes the tag probability, and (3) use an additional Bi-LSTM to consume the feature embeddings of each token and concatenate both Bi-LSTM encodings of feature embeddings and word embeddings before the output layer. We set the word input dimension to 100, word LSTM hidden layer dimension to 100, character input dimension to 50, character LSTM hidden layer dimension to 25, input dropout rate to 0.5, and use stochastic gradient descent with learning rate 0.01 for optimization.

5.3 Incorporating Non-traditional Linguistic Knowledge

In this section we will describe the detailed methods to acquire and encode various types of non-traditional resources. We call them as *non-traditional* because they have been rarely used in previous NLP research.

5.3.1 Basic Knowledge about the Language

Wikipedia Description. An English Wikipedia page about a language usually provides us general descriptions of the language. In particular, the list of usable characters,

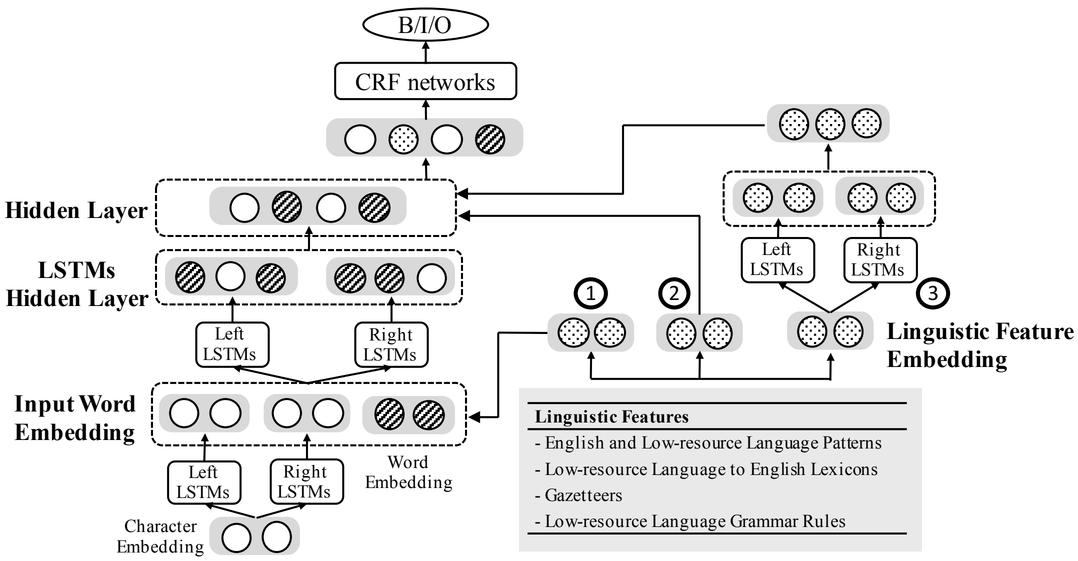


Figure 5.2: Three Integration Methods to Incorporate Explicit Linguistic Features into DNN.

gender indicators, capitalization information, transliteration and number spelling rules are most useful for name tagging. The list of usable characters for regular words in a particular language can help us detect foreign borrow words, which are likely to be names. For example,

"*th*" usually does not appear at the beginning of a Turkish word. Thus "*Thomas Marek*" is likely to be a foreign name.

Grammar Book. From grammar books we can also extract more language-specific contextual words, prefixes, suffixes and stemming rules. Name related lists contain: case suffix, preposition, postposition, ordinal number, definite article, negation, conjunction, pronoun, quantifier, numeral, time, locative, question particle, demonstrative, degree word, plural prefix/suffix, subordinator, reduplication, possessive, situational and epistemic markers. Table 5.1 shows some examples of name related suffix features.

5.3.2 Linguistic Structure

Recently linguists have made great efforts at building linguistic knowledge bases for thousands of languages in the world. Two such examples are WALS database [114] and Syntactic Structures of the World's Languages.[6] These databases classify languages according to a large number of topological properties (phonological, lexical and grammatical). For example, WALS consists of 141 maps with accompanying text on diverse properties, gathered

[6] http://sswl.railsplayground.net/

Languages	Features	Description	Examples
Uzbek	Name	**-ni** (accusative), **-ning** (possessive), **-da** (locative), **-dan** (ablative)	**Turkiyaning** (of Turkey), **Turkiyada** (in Turkey), **Turkiyaga** (to Turkey), **Turkiyadan** (from Turkey).
	Non-Name	Suffix **-roq** indicates adjectives	**qoraroq** (darker)
		Suffixes **-lar/-ler** indicate plurals	**qizlar** (daughters)
Hungarian	Name	Foreign name with >1 tokens and an adjective marker	New York-**i** (from New York)
		Most names with adjective or verbal suffix are lowercased	Balzac + **-os** ⇒ balzac**os**
		Possession relation	Péter-**ék** (Peter and his group), Péter-**é** (that of Peter)
		Affixes associated with names	Sartre-**nak** (to Sartre), Bordeaux-**ban** (in Bordeaux), Smith-**ért** (for Smith)
	Non-Name	Non-Name POS tag	adjectives (**-tlen**: "**-less**"), verbs tense (**meg-**:"completed"), conjunctions (**-ért**: "because of")
		Complete inflectional for nominals	karoknak (for arms) → karok (arms) → kar (arm)
Uyghur	Name	Animacy suffixes	**ning**, **ni**, **luq**, and **lik**
		Geopolitical or location suffixes	**ke**, **ge**, **qa**, **gha**, **te**, **de**, **ta**, **da**, **tin**, **din**, **tiki**, **diki**, **kiche**, **giche**, **qiche**, and **ghiche**.
Turkish	Name	Postpositions	karaköy**de** (in Karaköy)

Table 5.1: Name-related Knowledge Summarized from Grammar Books.

from descriptive materials (such as reference grammars).

Altogether there are 2,676 languages and more than 58,000 data points; each data point is a (language, feature, feature value) tuple that specifies the value of the feature in a particular language. (e.g., (English, canonical word order, SVO)). In total we extract 188 linguistic properties related to name tagging, belonging to 20 Phonology, 13 Lexicon, 12 Morphology, 29 Nominal, 8 Nominal Syntax, 17 Verbal Categories, 56 Word Order, 26 Simple Clauses, and 7 Complex Sentences categories respectively. Table 5.2 shows some examples.

5.3.3 Multi-lingual Dictionaries

CIA Names. We utilize the CIA Name Files,[7] which include biographical sketches, memorandums, telegrams, legislative records, legal documents, statements, and other records. We used the version cleaned up by Lawson et al.[8] that includes documents about names

[7] https://www.archives.gov/iwg/declassified-records/rg-263-cia-records

[8] https://www.researchgate.net/profile/Edwin_Lawson

Languages	Categories	Description	Name Related Characteristics
Tagalog	Subject, Verb, Object Order	VS, VO, VSO	the word at the beginning of a sentence is unlikely to be a name
Turkish	Negation	Suffix -me at the root of a verb indicates negations	not a name
Bengali	Animacy	-ta is a case that indicates inanimacy	
Thai	Nested Name Structure	Delimiter between modifier and head, [ORG กระทรวงต่างประเทศ] **ของ**[LOC อินโดนีเซีย] ([ORG Foreign Ministry] of [LOC Indonesia])	Name boundary
Tamil	Conjunction Structure	Name1-**yum** Name2-**yum** (Name1 and Name2)	Name type consistency

Table 5.2: Name-related Knowledge Extracted from WALS.

in 41 languages. Besides, person names in certain regions often include some common syllable patterns. Table 5.3 presents some examples. In languages such as Turkish, Uzbek and Uyghur, a person's last name inherits from his or her father's first name. In Uyghur, there are no additional suffixes. In Uzbek, additional suffixes include "*-ov*", "*-ev*", "*-yev*", "*-eva*" and "*-yeva*". In Turkish, a male's first name often ends with a consonant, and his last name consists of his father's first name and a suffix "*-oğlu* (son of)". We exploit this kind of knowledge to improve gazetteer match and name boundary identification.

Languages	Frequent Syllable Patterns	Examples
Slavic	Suffixes: -ov, -ev -ova, -eva; -ovich, -ich, -enko, -ko, -chuk, -yuk, -ak, -chenko, -skiy, -ski, -vych, -vich	Karimov, Yuriy Yarov, Abdulaziz Komilov, Yamonkulov Yaxshiboyevich, Shevchenko
Arabic	Prefixes: al-, Ahl, Abdul-, Abdu-	Abdul Khaliq, Abdul Latif, Abdul Maajid
	Suffixes: -allah, -ullah	Daifallah, Dhikrullah, Faizullah, Fathallah
Uzbek	Suffixes: -ov, -ova, -ev -yev, -eva, -yeva; -ovich, -evich, -ich	Karim Ahmedov, Ahmed Aliev, Zulfiya Karimova, Karmm Sharafovich Rashidov

Table 5.3: Common Syllable Patterns Extracted from CIA Names.

Unicode CLDR. Unicode Common Locale Data Repository (CLDR)[9] is a data collection for 194 languages, maintained by the Unicode Consortium to support software internationalization and localization. We extract bi-lingual location gazetteers, and exploit patterns and lists of currencies, months, weekdays, day periods and time units to remove them from

[9] http://cldr.unicode.org/

name candidates because they share some features with names (e.g., capitalization, "*Ocak*" in Turkish means "*January*").

Wiktionary. Wiktionary[10] is a web-based collaborative project to create an English content dictionary of all words in many languages. We collected dictionaries in 1,247 languages.

Panlex. Panlex[11] [115, 116] database contains 1.1 billion pairwise translations among 21 million expressions in about 10,000 language varieties.

Multilingual WordNet. We leverage three versions of multi-lingual WordNet: (1) Open Multilingual WordNet [117] which links words in many languages to English WordNet based on Wiktionary and CLDR; (2) Universal WordNet [118] which automatically extends English WordNet with around 1.5 million meaning links for 800,000 words in over 200 languages, based on WordNets, translation dictionaries and parallel corpora; and (3) Etymological WordNet [119, 120] that provides information about how words in various languages are etymologically related based on Wiktionary.

Phrase Pairs Mined from Wikipedia. From Wikipedia we extracted all pairs of titles that are connected by cross-lingual links. And we extracted more phrase translation pairs using parenthesis patterns from the beginning sentences of Wikipedia pages. For example, from the first sentence of the English Wikipedia page about Ürümqi: "*Ürümqi (ئۈرۈمچى) is the capital of the Xinjiang Uyghur Autonomous Region of the People's Republic of China in Northwest China,*" we can extract an Uyghur-English name translation pair of "ئۈرۈمچى" and "*Ürümqi*". Moreover, we retrieved related Wikipedia articles, and mined common names in many languages and regions.

GeoNames. We exploit the geo-political and location entities in multilingual GeoNames database.[12] It contains over 10 million geographical names and over 9 million unique features of the following properties: id, name, asciiname, alternate names, latitude, longitude, feature class, feature code, country code, administrative code, population, elevation and time zone.

JRC Names. Finally we include the JRC Names [121], a large list of person and organization names (about 205,000 entries) in over 20 different scripts. Some entries include additional information such as frequency, title and date ranges.

[10] https://en.wiktionary.org

[11] http://panlex.org/

[12] http://www.geonames.org/

Language	**Gazetteer**			**Title**	**Non-Name**	**Suffix**
	PER	LOC	ORG			
Hausa	1,174	5,123	199	42	391	21
Turkish	2,819	7,271	262	231	411	181
Uzbek	1,771	5,331	103	178	271	209

Table 5.4: Name Related List Statistics (# of entries).

5.3.4 Encoding Linguistic Features

We merged the linguistic resources collected above into three types of features: (1) name gazetteers; (2) list of suffixes and contextual words (e.g., titles) that indicate names; and (3) list of words that indicate non-names (e.g., time expressions). Ultimately we obtained 30 explicit linguistic feature categories. Table 5.4 shows the statistics of the encoded features.

For each token w_i in a sentence, we check whether w_i, its previous token w_{i-1} and its next token w_{i+1} exist in these lists, and concatenate them into an initial feature vector for w_i. For any resources (e.g., lexicons and phrase books) that contain English translations, we also use them to translate each w_i, and check whether its translation is capitalized or exists in English name tagging resources (contextual words, gazetteers), whether its contexts match any English patterns as described in [37].

5.4 Experiments

Using the data sets mentioned in Section 5.2.1, we conduct experiments for three languages: Hausa, Turkish and Uzbek.

Table 5.5 compares the results of three feature integration methods described in Section 5.2.2 and Figure 5.2. We can see that the third integration method (Integration 3) consistently outperforms the others for all three languages.

Models	**Hausa**	**Turkish**	**Uzbek**
Bi-LSTMs	65.7	65.9	64.1
+ Integration 1	71.1	71.8	67.4
+ Integration 2	71.5	73.1	67.2
+ Integration 3	**72.2**	**74.3**	**68.4**

Table 5.5: Feature Integration Methods Comparison.

We compare the following models: a baseline model that uses only character and word embedding features, a model adding traditional linguistic features as described in [37], and a model further adding non-traditional linguistic features using the third integration method. Figure 5.3 presents the results. Clearly models trained with linguistic features substantially outperform the baseline models on all noise levels for all languages. As the noise level increases, the performance of the baseline model drops drastically while the model trained with linguistic features successfully curbs the downward trend and forms a relatively flat curve at last. Adding non-traditional linguistic features provides further gains in almost all settings. Notably for Turkish, adding linguistic features and using 100% automatically generated noisy training data, our approach achieves the same performance as the baseline model using 75% manually created clean data and 25% automatically created noisy data. In other words, explicit linguistic knowledge has significantly saved annotation cost (2,367 sentences). Our results without using any manually labeled training data are much better than state-of-the-art reported in our previous work [37] which used most traditional resources mentioned in this paper and [85] which derived noisy training data from Wikipedia markups. On the same test sets we achieved 5.5% higher F-score for Hausa than [37], 27.7% higher F-score for Turkish and 13.6% higher F-score for Uzbek than [85].

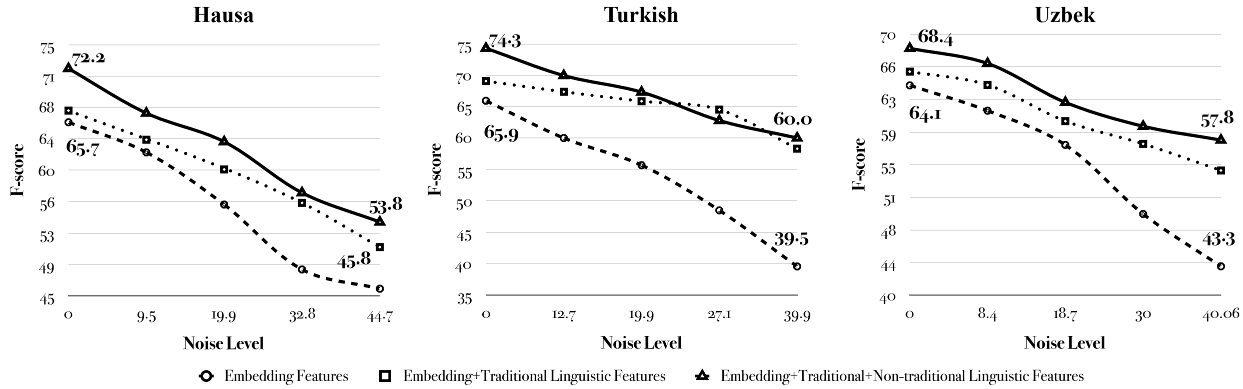
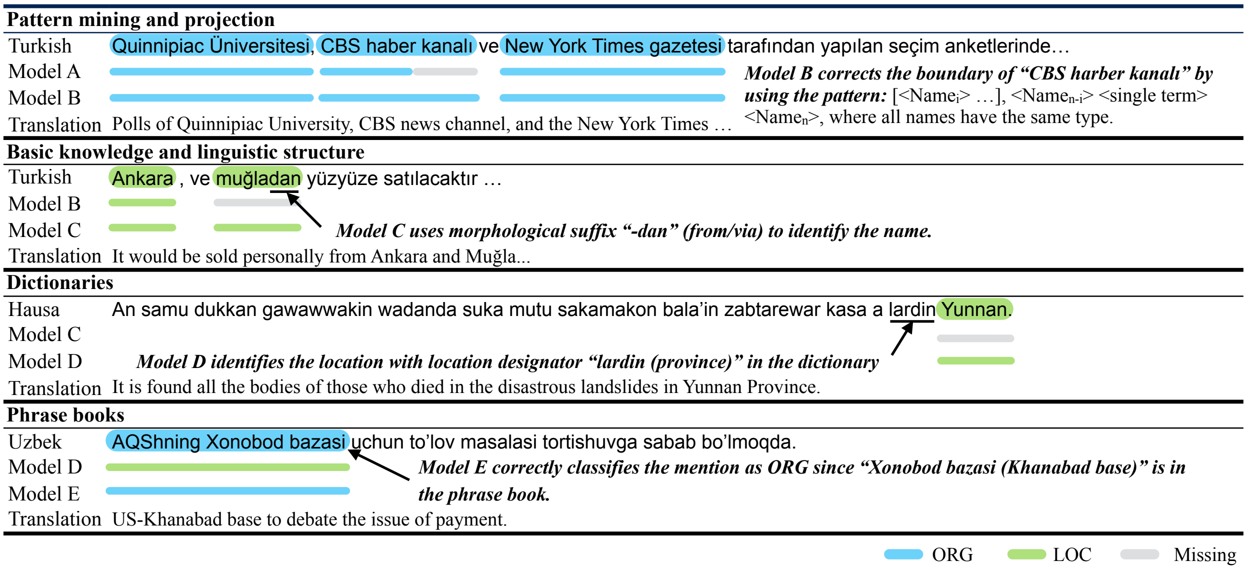


Figure 5.3: Name Tagging Performance.

Table 5.6 presents the contribution of each linguistic feature category when using 100% automatically created training data. Figure 5.4 shows some examples of errors corrected by each category.

	Category	Hausa	Turkish	Uzbek
A	Embedding feature	45.8	39.5	43.3
B	(A)+Pattern mining and projection	46.7	40.9	45.4
C	(B)+Basic knowledge and linguistic structure	50.4	53.3	52.4
D	(C)+Dictionaries	52.0	57.7	56.1
E	(D)+Phrase books	53.8	60.0	57.8

Table 5.6: Contributions of Various Categories of Linguistic Knowledge (F-score (%)).



Pattern mining and projection

Turkish Quinnipiac Üniversitesi, CBS haber kanalı ve New York Times gazetesi tarafından yapılan seçim anketlerinde…

Model A

Model B

Model B corrects the boundary of "CBS harber kanalı" by using the pattern: [<Name$_i$> …], <Name$_{n-i}$> <single term> <Name$_n$>, where all names have the same type.

Translation Polls of Quinnipiac University, CBS news channel, and the New York Times …

Basic knowledge and linguistic structure

Turkish Ankara , ve muğladan yüzyüze satılacaktır …

Model B

Model C

Model C uses morphological suffix "-dan" (from/via) to identify the name.

Translation It would be sold personally from Ankara and Muğla...

Dictionaries

Hausa An samu dukkan gawawwakin wadanda suka mutu sakamakon bala'in zabtarewar kasa a lardin Yunnan.

Model C

Model D ***Model D identifies the location with location designator "lardin (province)" in the dictionary***

Translation It is found all the bodies of those who died in the disastrous landslides in Yunnan Province.

Phrase books

Uzbek AQShning Xonobod bazasi uchun to'lov masalasi tortishuvga sabab bo'lmoqda.

Model D

Model E

Model E correctly classifies the mention as ORG since "Xonobod bazasi (Khanabad base)" is in the phrase book.

Translation US-Khanabad base to debate the issue of payment.

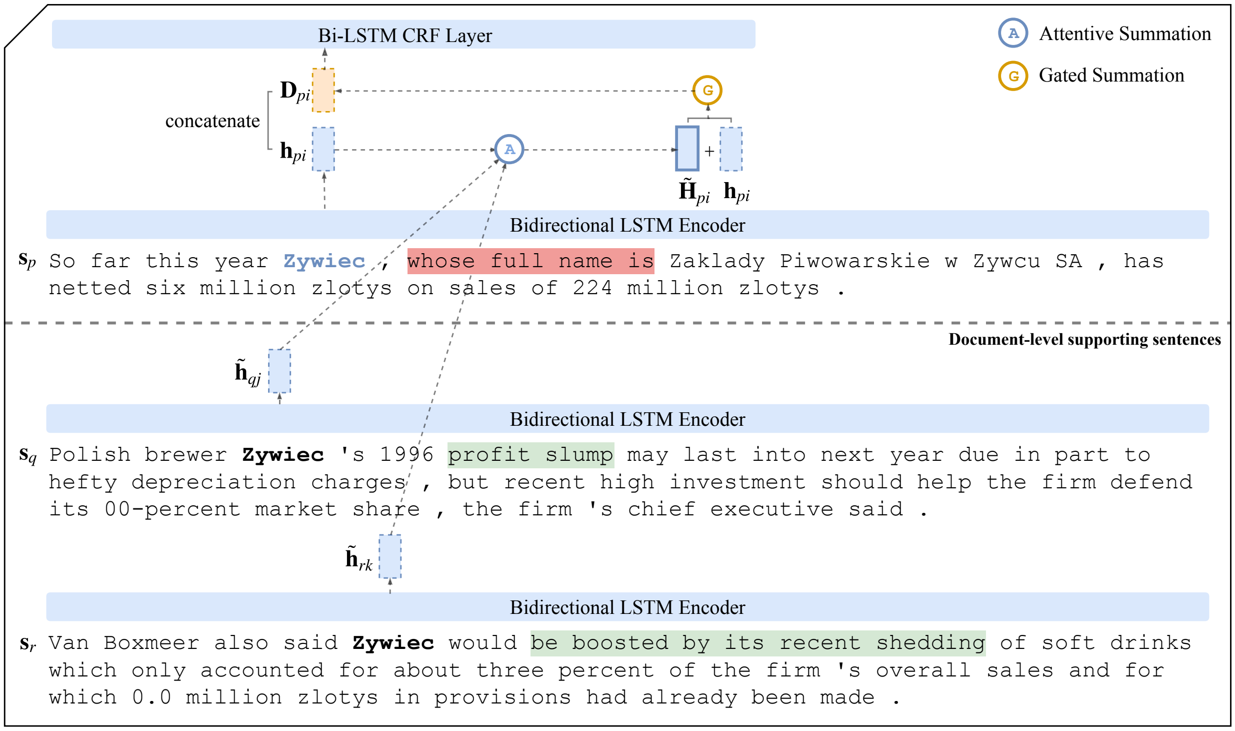
Figure 5.4: Examples of Corrections Made by Each Category of Linguistic Knowledge.

Chapter 6
Global Attention for Name Tagging

6.1 Approach Overview

Local context (*i.e.*, surrounding tokens) is crucial for name tagging because the context gives insight to the semantic meaning of the token. However, in cases where the local context is ambiguous or lacks sufficient content, the classifier would fail to make correct predictions. In this section, we propose the document-level, and corpus-level attention to incorporate external contextual information to address this challenge.

6.1.1 Document-level Attention



Bi-LSTM CRF Layer
A Attentive Summation
G Gated Summation
$\mathbf{D}_{pi}$
concatenate
$\mathbf{h}_{pi}$
A
G
$\tilde{\mathbf{H}}_{pi}$
+
$\mathbf{h}_{pi}$
Bidirectional LSTM Encoder
$\mathbf{s}_p$
Document-level supporting sentences
$\tilde{\mathbf{h}}_{qj}$
Bidirectional LSTM Encoder
$\mathbf{s}_q$
$\tilde{\mathbf{h}}_{rk}$
Bidirectional LSTM Encoder
$\mathbf{s}_r$

So far this year **Zywiec** , whose full name is Zaklady Piwowarskie w Zywcu SA , has netted six million zlotys on sales of 224 million zlotys .

Polish brewer **Zywiec** 's 1996 profit slump may last into next year due in part to hefty depreciation charges , but recent high investment should help the firm defend its 00-percent market share , the firm 's chief executive said .

Van Boxmeer also said **Zywiec** would be boosted by its recent shedding of soft drinks which only accounted for about three percent of the firm 's overall sales and for which 0.0 million zlotys in provisions had already been made .

Figure 6.1: Document-level Attention Architecture. (Within-sequence context in red incorrectly indicates the name as PER, and document-level context in green correctly indicates the name as ORG.)

Many entity mentions are tagged as multiple types by the baseline approach within the same document due to ambiguous contexts (14.43% of the errors in English, 18.55%

in Dutch, and 17.81% in German). This type of error is challenging to address as most of the current neural network based approaches focus on evidence within the sequence when making decisions. In cases where a sentence is short or highly ambiguous, the model may either fail to identify names due to insufficient information or make wrong decisions by using noisy context. In contrast, a human in this situation may seek additional evidence from other sentences within the same document to improve judgments [122].

In Figure 1.2, the baseline model mistakenly tags "`Zywiec`" as PER due to the ambiguous context "`whose full name is`...", which frequently appears around a person's name. However, contexts from other sentences in the same document containing "`Zywiec`" (*e.g.,* s_q and s_r in Figure 6.1), such as "`'s 1996 profit`..." and "`would be boosted by its recent shedding`...", indicate that "`Zywiec`" ought to be tagged as ORG. Thus, we incorporate the document-level supporting evidence with the following attention mechanism [123].

Formally, given a document $D = \{s_1, s_2, ...\}$, where $s_i = \{w_{i1}, w_{i2}, ...\}$ is a sequence of words, we apply a Bi-LSTM to each word in s_i, generating local contextual representations $h_i = \{\mathbf{h}_{i1}, \mathbf{h}_{i2}, ...\}$. Next, for each w_{ij}, we retrieve the sentences in the document that contain w_{ij} (*e.g.,* s_q and s_r in Figure 6.1) and select the local contextual representations of w_{ij} from these sentences as supporting evidence, $\tilde{h}_{ij} = \{\tilde{\mathbf{h}}^1_{ij}, \tilde{\mathbf{h}}^2_{ij}, ...\}$ (*e.g.,* $\tilde{\mathbf{h}}_{qj}$ and $\tilde{\mathbf{h}}_{rk}$ in Figure 6.1), where h_{ij} and $\tilde{h}_{ij}$ are obtained with the same Bi-LSTM. Since each representation in the supporting evidence is not equally valuable to the final prediction, we apply an attention mechanism to weight the contextual representations of the supporting evidence:

$$e^k_{ij} = \mathbf{v}^\top \tanh\left(W_h \mathbf{h}_{ij} + W_{\tilde{h}} \tilde{\mathbf{h}}^k_{ij} + \mathbf{b}_e\right) \ ,$$

$$\alpha^k_{ij} = \text{Softmax}\left(e^k_{ij}\right) \ ,$$

where $\mathbf{h}_{ij}$ is the local contextual representation of word j in sentence s_i and $\tilde{\mathbf{h}}^k_{ij}$ is the k-th supporting contextual representation. W_h, $W_{\tilde{h}}$ and $\mathbf{b}_e$ are learned parameters. We compute the weighted average of the supporting representations by

$$\tilde{\mathbf{H}}_{ij} = \sum_{k=1} \alpha^k_{ij} \tilde{\mathbf{h}}^k_{ij} \ ,$$

where $\tilde{\mathbf{H}}_{ij}$ denotes the contextual representation of the supporting evidence for w_{ij}.

For each word w_{ij}, its supporting evidence representation, $\tilde{\mathbf{H}}_{ij}$, provides a summary of the other contexts where the word appears. Though this evidence is valuable to the prediction process, we must mitigate the influence of the supporting evidence since the prediction should still be made primarily based on the query context. Therefore, we apply a gating mechanism to constrain this influence and enable the model to decide the amount of the supporting evidence that should be incorporated in the prediction process, which is given by

$$\mathbf{r}_{ij} = \sigma(W_{\tilde{H},r}\tilde{\mathbf{H}}_{ij} + W_{h,r}\mathbf{h}_{ij} + \mathbf{b}_r) \ ,$$

$$\mathbf{z}_{ij} = \sigma(W_{\tilde{H},z}\tilde{\mathbf{H}}_{ij} + W_{h,z}\mathbf{h}_{ij} + \mathbf{b}_z) \ ,$$

$$\mathbf{g}_{ij} = \tanh(W_{h,g}\mathbf{h}_{ij} + \mathbf{z}_{ij} \odot (W_{\tilde{H},g}\tilde{\mathbf{H}}_{ij} + \mathbf{b}_g)) \ ,$$

$$\mathbf{D}_{ij} = \mathbf{r}_{ij} \odot \mathbf{h}_{ij} + (1 - \mathbf{r}_{ij}) \odot \mathbf{g}_{ij} \ ,$$

where all W, $\mathbf{b}$ are learned parameters and $\mathbf{D}_{ij}$ is the gated supporting evidence representation for w_{ij}.

6.1.2 Topic-aware Corpus-level Attention

The document-level attention fails to generate supporting evidence when the name appears only once in a single document. In such situations, we analogously select supporting sentences from the entire corpus. Unfortunately, different from the sentences that are naturally topically relevant within the same documents, the supporting sentences from the other documents may be about distinct topics or scenarios, and identical phrases may refer to various entities with different types, as in the example in Figure 1.2.

To narrow down the search scope from the entire corpus and avoid unnecessary noise, we introduce a topic-aware corpus-level attention which clusters the documents by topic and carefully selects topically related sentences to use as supporting evidence.

We first apply Latent Dirichlet allocation (LDA) [124] to model the topic distribution of each document and separate the documents into N clusters based on their topic distributions.[13] As in Figure 6.2, we retrieve supporting sentences for each word, such as "`Zywiec`", from the topically related documents and employ another attention mechanism [123] to the supporting contextual representations, $\hat{h}_{ij} = \{\hat{\mathbf{h}}^1_{ij}, \hat{\mathbf{h}}^2_{ij}, ...\}$ (*e.g.*, $\tilde{\mathbf{h}}_{xi}$ and $\tilde{\mathbf{h}}_{yi}$ in Figure 6.2). This yields a weighted contextual representation of the corpus-level supporting evidence, $\hat{\mathbf{H}}_{ij}$, for each w_{ij}, which is similar to the document-level supporting evidence representation,

[13] $N = 20$ in our experiments.

$\tilde{\mathbf{H}}_{ij}$, described in section 6.1.1. We use another gating mechanism to combine $\hat{\mathbf{H}}_{ij}$ and the local contextual representation, $\mathbf{h}_{ij}$, to obtain the corpus-level gated supporting evidence representation, $\mathbf{C}_{ij}$, for each w_{ij}.

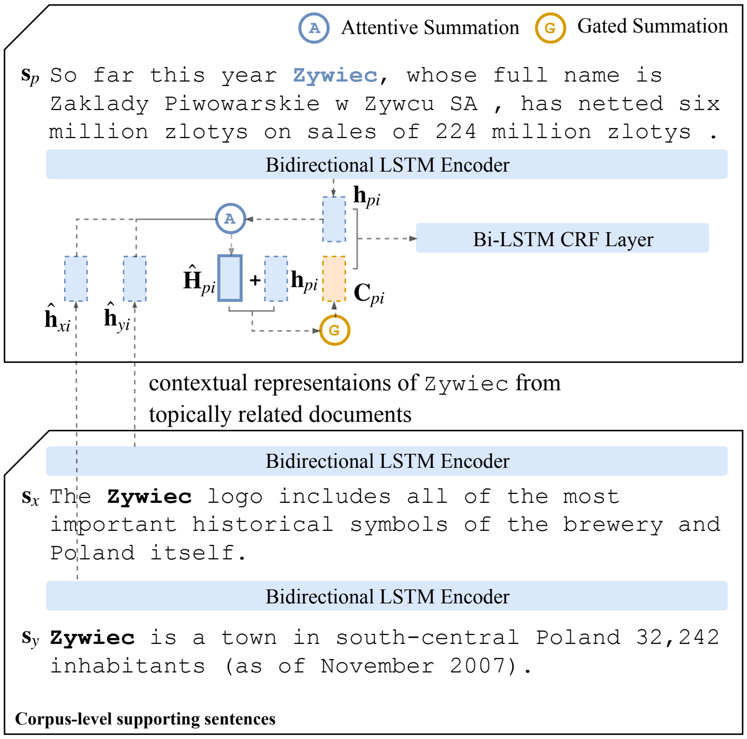

A Attentive Summation
G Gated Summation
$\mathbf{s}_p$ So far this year Zywiec, whose full name is Zaklady Piwowarskie w Zywcu SA , has netted six million zlotys on sales of 224 million zlotys .
Bidirectional LSTM Encoder
$\mathbf{h}_{pi}$
Bi-LSTM CRF Layer
$\hat{\mathbf{H}}_{pi}$
+
$\mathbf{h}_{pi}$
$\mathbf{C}_{pi}$
$\hat{\mathbf{h}}_{xi}$
$\hat{\mathbf{h}}_{yi}$
contextual representaions of Zywiec from topically related documents
Bidirectional LSTM Encoder
$\mathbf{s}_x$ The Zywiec logo includes all of the most important historical symbols of the brewery and Poland itself.
Bidirectional LSTM Encoder
$\mathbf{s}_y$ Zywiec is a town in south-central Poland 32,242 inhabitants (as of November 2007).
Corpus-level supporting sentences

Figure 6.2: Corpus-level Attention Architecture.

6.1.3 Tag Prediction

For each word w_{ij} of sentence s_i, we concatenate its local contextual representation $\mathbf{h}_{ij}$, document-level gated supporting evidence representation $\mathbf{D}_{ij}$, and corpus-level gated supporting evidence representation $\mathbf{C}_{ij}$ to obtain its final representation. This representation is fed to another Bi-LSTM to further encode the supporting evidence and local contextual features into an unified representation, which is given as input to an affine-CRF layer for label prediction.

6.2 Experiments

6.2.1 Dataset

We evaluate our methods on the CoNLL-2002 and CoNLL-2003 name tagging datasets [125]. The CoNLL-2002 dataset contains name tagging annotations for Dutch (NLD) and Spanish

(ESP), while the CoNLL-2003 dataset contains annotations for English (ENG) and German (DEU). Both datasets have four pre-defined name types: person (PER), organization (ORG), location (LOC) and miscellaneous (MISC).[14]

Code	**Train**	**Dev.**	**Test**
NLD	202,931 (13,344)	37,761 (2,616)	68,994 (3,941)
ESP	264,715 (18,797)	52,923 (4,351)	51,533 (3,558)
ENG	204,567 (23,499)	51,578 (5,942)	46,666 (5,648)
DEU	207,484 (11,651)	51,645 (4,669)	52,098 (3,602)

Table 6.1: # of tokens in name tagging datasets statistics. # of names is given in parentheses.

We select at most 4 document-level supporting sentences and 5 corpus-level supporting sentences.[15] Since the document-level attention method requires input from sets of documents, we do not evaluate the document-level attention on the CoNLL-2002 Spanish dataset which lacks document delimiters. We still evaluate the corpus-level attention on the Spanish dataset by randomly splitting the dataset into documents (30 sentences per document). Although randomly splitting the sentences does not yield perfect topic modeling clusters, experiments show the corpus-level attention still outperforms the baseline (Section 6.2.3).

6.2.2 Experimental Setup

For word representations, we use 100-dimensional pre-trained word embeddings and 25-dimensional randomly initialized character embeddings. We train word embeddings using the word2vec package.[16] English embeddings are trained on the English Giga-word version 4, which is the same corpus used in [59]. Dutch, Spanish, and German embeddings are trained on corresponding Wikipedia articles (2017-12-20 dumps). Word embeddings are fine-tuned during training.

Table 6.2 shows our hyper-parameters. For each model with an attention, since the Bi-LSTM encoder must encode the local, document-level, and/or corpus-level contexts, we pre-train a Bi-LSTM CRF model for 50 epochs, add our document-level attention and/or corpus-level attention, and then fine-tune the augmented model. Additionally, [126] report

[14] The miscellaneous category consists of names that do not belong to the other three categories.

[15] Both numbers are tuned from 1 to 10 and selected when the model performs best on the development set.

[16] https://github.com/tmikolov/word2vec

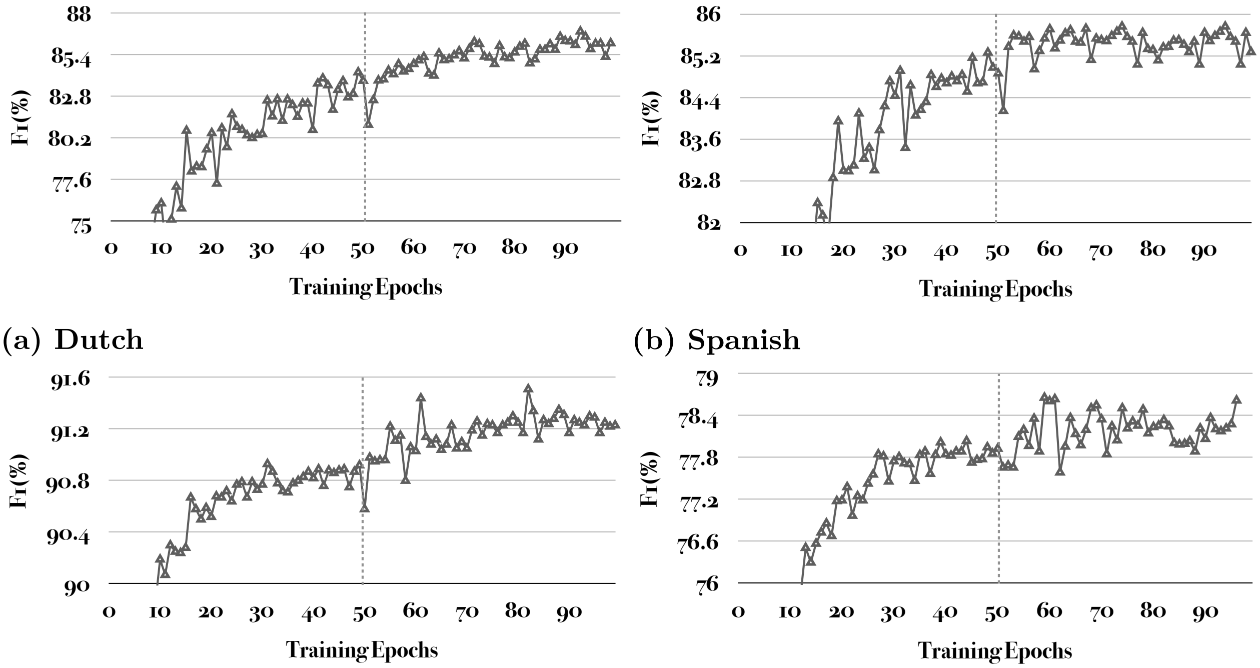
Hyper-parameter	Value
CharCNN Filter Number	25
CharCNN Filter Widths	[2, 3, 4]
Lower Bi-LSTM Hidden Size	100
Lower Bi-LSTM Dropout Rate	0.5
Upper Bi-LSTM Hidden Size	100
Learning Rate	0.005
Batch Size	N/A∗
Optimizer	SGD [127]

∗ Each batch is a document. The batch size varies as the different document length.

Table 6.2: Hyper-parameters.

that neural models produce different results even with same hyper-parameters due to the variances in parameter initialization. Therefore, we run each model ten times and report the mean as well as the maximum F1 scores.

6.2.3 Performance Comparison



(a) Dutch

(b) Spanish

(c) English

(d) German

Figure 6.3: Average F1 score for each epoch of the ten runs of our model with both document-level and corpus-level attentions. Epochs 1-50 are the pre-training phase and 51-100 are the fine-tuning phase.

We compare our methods to three categories of baseline name tagging methods:

- **Vanilla Name Tagging** Without any additional resources and supervision, the current state-of-the-art name tagging model is the Bi-LSTM-CRF network reported by [59] and [60], whose difference lies in using a LSTM or CNN to encode characters. Our methods fall in this category.
- **Multi-task Learning** [129, 63] apply multi-task learning to boost name tagging performance by introducing additional annotations from related tasks, such as entity linking and POS tagging labels.
- **Join-learning with Language Model** [30, 61, 29] leverage a pre-trained language model on a large external corpus to enhance the semantic representations of words in the local corpus. [29] achieves a remarkably high score on the CoNLL-2003 English dataset using a giant language model pre-trained on a 1 Billion Word Benchmark [130].

Table 6.3 presents the performance comparison between the baseline, the aforementioned state-of-the-art methods, and our proposed methods. Adding only the document-level attention offers a F1 gain of between 0.37% and 1.25% on Dutch, English, and German. Similarly, the addition of the corpus-level attention yields a F1 gain between 0.46% to 1.08% across all four languages. The model with both attentions outperforms our baseline method by 1.60%, 0.56%, and 0.79% on Dutch, English, and German, respectively.

By incorporating the document-level and corpus-level attentions, we achieve state-of-the-art performance on the Dutch (NLD), Spanish (ESP) and German (DEU) datasets. For English, our methods outperform the state-of-the-art methods in the "Vanilla Name Tagging" category. Since the document-level and corpus-level attentions introduce redundant and topically related information, our models are compatible with the language model enhanced approaches. It is interesting to explore the integration of these two methods, but we leave this to future explorations. Figure 6.3 presents, for each language, the learning curves of the full models (*i.e.,* with both document-level and corpus-level attentions). The learning curve is computed by averaging the F1 scores of the ten runs at each epoch. We first pre-train a baseline Bi-LSTM CRF model from epoch 1 to 50. Then, starting at epoch 51, we incorporate the document-level and corpus-level attentions to fine-tune the entire model. As shown in Figure 6.3, when adding the attentions at epoch 51, the F1 score drops significantly as new parameters are introduced to the model. The model gradually adapts to the new information, the F1 score rises, and the full model eventually outperforms the pre-trained model. The learning curves strongly prove the effectiveness of our proposed methods.

6.2.4 Qualitative Analysis

#1 Dutch	
Baseline	`[B-LOC Granada]` overwoog vervolgens een bod op Carlton uit te brengen, maar daar ziet het concern nu van af. *Granada then considered issuing a bid for Carlton, but the concern now sees it.*
Our model	`[B-ORG Granada]` overwoog vervolgens een bod op Carlton uit te brengen, maar daar ziet het concern nu van af.
D-lvl sentences	`[B-ORG Granada]` `[I-ORG Media]` neemt belangen in United News. *Granada Media takes interests in United News.*
C-lvl sentences	Het Britse concern `[B-ORG Granada]` `[I-ORG Media]` heeft voor 1,75 miljard pond sterling (111 miljard Belgische frank) aandelen gekocht van United News Media. *The British group Granada Media has bought shares of GBP 1.75 trillion (111 billion Belgian francs) from United News Media.*
#2 English	
Baseline	Initially Poland offered up to 75 percent of Ruch but in March `[ORG Kaczmarek]` cancelled the tender and offered a minority stake with an option to increase the equity.
Our model	Initially Poland offered up to 75 percent of Ruch but in March `[PER Kaczmarek]` cancelled the tender and offered a minority stake with an option to increase the equity.
D-lvl sentences	`[PER Kaczmarek]` said in May he was unhappy that only one investor ended up bidding for Ruch.
#3 German	
Baseline	Diese Diskussion werde ausschlaggebend sein für die Stellungnahme der **Grünen** in dieser Frage. *This discussion will be decisive for the opinion of the Greens on this question.*
Our model	Diese Diskussion werde ausschlaggebend sein für die Stellungnahme der `[B-ORG Grünen]` in dieser Frage.
C-lvl sentences	Auch das Mitglied des Bundesvorstandes der `[B-ORG Grünen]`, Helmut Lippelt, sprach sich für ein Berufsheer au. *Helmut Lippelt, a member of the Federal Executive of the Greens, also called for a professional army.*
#4 Negative Example	
Reference	`[B-LOC Indianapolis]` 1996-12-06
Our model	`[B-ORG Indianapolis]` 1996-12-06
D-lvl sentence	The injury-plagued `[B-ORG Indianapolis]` `[I-ORG Colts]` lost another quarterback on Thursday but last year's AFC finalists rallied together to shoot down the Philadelphia Eagles 37-10 in a showdown of playoff contenders.

* D-lvl sentences: document-level supporting sentences.

* C-lvl sentences: corpus-level supporting sentences.

Figure 6.4: Comparison of name tagging results between the baseline and our methods.

Table 6.4 compares the name tagging results from the baseline model and our best models. All examples are selected from development set.

In the Dutch example, "`Granada`" is the name of a city in Spain, but also the short name of "`Granada Media`". Without ORG related context, "`Granada`" is mistakenly tagged as LOC by the baseline model. However, the document-level and corpus-level supporting evidence retrieved by our method contains the ORG name "`Granada Media`", which strongly indicates "`Granada`" to be an ORG in the query sentence. By adding the document-level and corpus-level attentions, our model successfully tags "`Granada`" as ORG.

In example 2, the OOV word "`Kaczmarek`" is tagged as ORG in the baseline output. In the retrieved document-level supporting sentences, PER related contextual information, such as the pronoun "`he`", indicates "`Kaczmarek`" to be a PER. Our model correctly tags "`Kaczmarek`" as PER with the document-level attention.

In the German example, "`Grünen`" (Greens) is an OOV word in the training set. The character embedding captures the semantic meaning of the stem "`Grün`" (Green) which is a common non-name word, so the baseline model tags "`Grünen`" as O (outside of a name). In contrast, our model makes the correct prediction by incorporating the corpus-level attention because in the related sentence from the corpus "`Bundesvorstandes der Grünen`" (Federal Executive of the Greens) indicates "`Grünen`" to be a company name.

Code	Model	F1 (%)	
NLD	Gillick et al., 2015 [128]	reported	82.84
	Lample et al., 2016 [59]	reported	81.74
	Yang et al., 2017 [63]	reported	85.19
	Our Baseline	mean	85.43
		max	85.80
	Doc-lvl Attention	mean	86.82
		max	87.05
	Corpus-lvl Attention	mean	86.41
		max	86.88
	Both	mean	87.14
		max	**87.40**
		Δ	**+1.60**
ESP	Gillick et al., 2015 [128]	reported	82.95
	Lample et al., 2016 [59]	reported	85.75
	Yang et al., 2017 [63]	reported	85.77
	Our Baseline	mean	85.33
		max	85.51
	Corpus-lvl Attention	mean	85.77
		max	**86.01**
		Δ	**+0.50**
ENG	Luo et al., 2015 [129]	reported	91.20
	Lample et al., 2016 [59]	reported	90.94
	Ma and Hovy, 2016 [60]	reported	91.21
	Liu et al., 2017 [61]	reported	91.35
	Peters et al., 2017 [30]	reported	91.93
	Peters et al., 2018[29]	reported	**92.22**
	Our Baseline	mean	90.97
		max	91.23
	Doc-lvl Attention	mean	91.43
		max	91.60
	Corpus-lvl Attention	mean	91.41
		max	91.71
	Both	mean	91.64
		max	**91.81**
		Δ	**+0.58**
DEU	Gillick et al., 2015 [128]	reported	76.22
	Lample et al., 2016 [59]	reported	78.76
	Our Baseline	mean	78.15
		max	78.42
	Doc-lvl Attention	mean	78.90
		max	79.19
	Corpus-lvl Attention	mean	78.53
		max	78.88
	Both	mean	78.83
		max	**79.21**
		Δ	**+0.79**

Table 6.3: Performance of our methods versus the baseline and state-of-the-art models.

Chapter 7
Cross-lingual Language Model for Name Tagging

In this chapter, we aim to address the problem of training an LL name tagger without using any LL labeled data. Our proposed approaches are based on the following hypotheses introduced in section 1.3:

> ***Distributed word embeddings/contextualized word embeddings of HL and LL can be projected into a shared space, so that machine learning models trained on HL embeddings/contextualized embeddings can produce satisfying performance on LL.***

We first present our baseline method where we train a cross-lingual name tagger using cross-lingual word embeddings, and then we propose a new method to unsupervisedly pre-train cross-lingual language model for HL and LL via monolingual corpora. In the experiment section, we show cross-lingual name tagger trained with cross-lingual language model significantly outperforms cross-lingual word embeddings.

7.1 Approach Overview

In this section, We present the details of training cross-lingual word embeddings and cross-lingual language model, as well as how we use these embeddings to develop a cross-lingual name tagger.

7.1.1 Cross-lingual Word Embeddings

First we use Fasttext [131] to pre-train word embeddings on HL and LL Wikipedia corpus. Then we apply MUSE [26] toolkit to unsupervisedly align HL and LL monolingual embeddings to generate the HL-LL cross-lingual word embeddings $W_{XL} = W_{HL} \bigcup W_{LL}$, where W_{HL} is the HL portions of the cross-lingual word embeddings, and W_{LL} is the LL portions.

We choose the widely used Bi-LSTM CRF as our name tagging architecture. Given HL labeled training data, we train a HL name tagger by initializing its word embeddings as W_{HL}. We freeze word embeddings during training. When evaluating on LL test set, we encode LL sentence with W_{LL} and then feed them into the HL name tagger.

In addition to the unsupervised approach (HL labeled data is considered as existing resources), when a small amount of LL labeled data is available, we propose a new framework to utilize both HL and LL labeled data. Similarly, we first train a Bi-LSTM CRF HL name tagger, and then we finetune the name tagger on the LL labeled training data. As character representations are effective for name tagging [59, 60], we concatenate a randomly initialized CNN character representation to generate the word representation during finetuning. Similarly, we freeze word embedding update during finetuning.

7.1.2 Cross-lingual Language Model

Contextualized word embedding pre-training has shown to be effective for improving many NLP tasks [27, 28, 29, 30, 31, 32]. [27] completely remove pre-trained word embeddings and achieve the state-of-the-art performance by only feeding a pre-trained transformer language model to a Bi-LSTM name tagger. This somewhat proves pre-trained language model can produce better word representations over word embeddings.

Generally, training cross-lingual language model consists of two phases: 1) pre-train HL and LL encoder and decoder, and 2) apply "back-translation" to constrain the latent representations produced by the encoder to be shared across HL and LL. We iteratively repeat these two steps for each batch of training instances.

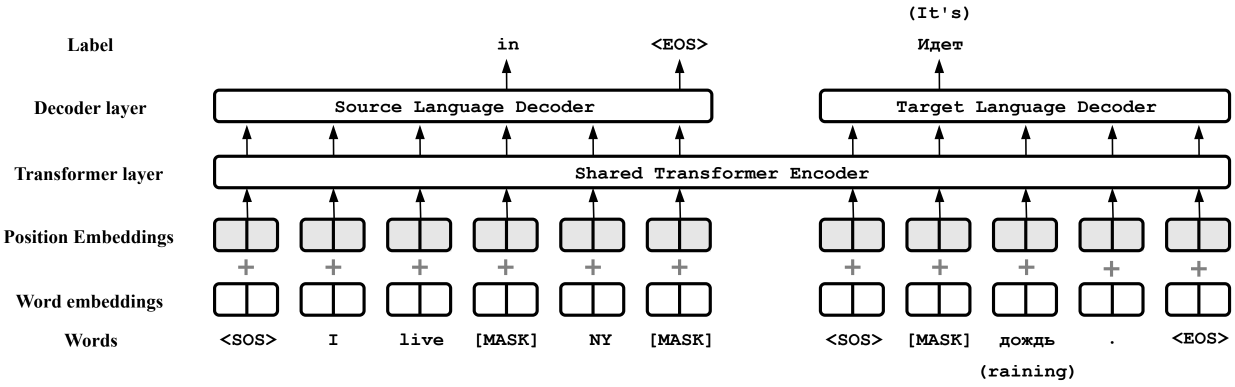


Figure 7.1: An example of training HL and LL language model.

Phase 1 - Figure 7.1 presents the architecture to train HL and LL language model. Following [27], we use Masked Language Modeling where we first randomly sample 15% of the words from the sentence, and then replace 80% of the sampled words with a mask token "[MASK]", 10% of them with a random token, and keep the rest 10% unchanged. The purpose of keeping 10% of words unchanged is to bias the representation towards the actual observed word [27]. The encoder is shared to take the most advantage of overlapped vocabulary between HL and LL, but the decoders are independent from each other.

Phase 2 - [34, 132] successfully use “back-translation” to unsupervisedly train a machine translation system and achieve remarkable performance, given no parallel data provided. Their decoder is an attentive RNN that produces various length sentence, while our decoder is a feed forward network with a softmax layer that predicts the masked word for each input token. Our decoder is more appropriate for name tagging task.

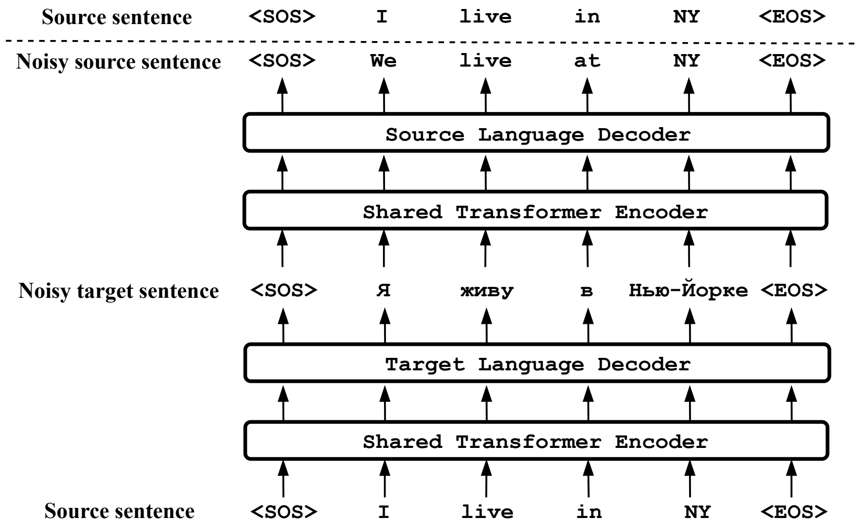


Figure 7.2: An example of back-translation.

Figure 7.2 shows an example of “back-translation” for English and Russian. Given an English sentence “I live in NY .”, the latent representations produced by the encoder are fed into a target language (Russian in this case) decoder which outputs a noisy Russian translation. Then we use the encoder again to generate latent representations of the Russian translation. A source language (English in this case) decoder decodes the latent representations back to English. As these two latent representations represent the same meaning, we use the original English sentence “I live in NY .” as labels of the source language decoder output. For parameter updates, we freeze the encoder and word embeddings, and only change source and tagert language decoders.

We simply add a feed-forward network and softmax layer on top of the encoder, and train the model on HL name tagging annotations. The encoder parameters are fixed during training. When applying to LL, we feed the encoder with LL words.

Byte Pair Encoding (BPE) - [27, 133] report subword encoding significantly improve the quality of language model pre-training and machine translation. We use fastBPE[17] to pre-process HL and LL monolingual data. The subword vocabularies of HL and LL are shared during training.

7.2 Experiments

In this section, we present the dataset, training details, and cross-lingual name tagging results on English-Russian and English-Spanish language pairs using cross-lingual word embeddings and language model.

7.2.1 Dataset

Language	**Train**	**Dev**	**Test**
English	14,987	N/A	N/A
Russian	4,755	1,953	1,953
Spanish	8,323	1,914	1,516

Table 7.1: # of sentences in name tagging datasets statistics.

In our experiments, we consider English as HL, and Russian and Spanish as LL. Spanish is a language close to English because its vocabulary and character set highly overlap with English, while Russian is a distant language compared to English due to its distinct vocabulary and characters. We use English CoNLL-2003 [125] for name tagger training, and evaluate it on Russian test set from DARPA LORELEI program (LDC2016E95) and Spanish CoNLL-2002 test set. For comparison, we use the name taggers trained on LL training set as the upper bound of our cross-lingual name taggers. Table 7.1 shows data statistics. To pre-train cross-lingual word embeddings and language model, we use English, Russian and Spanish 20190201 Wikipedia dump.

7.2.2 Training Details

Table 7.2 shows our hyper-parameters and training setup. We implement our cross-lingual language model in Pytorch, and train it on a 5 Tesla P100 GPUs.

[17] https://github.com/glample/fastBPE

Hyper-parameter	**Value**
Cross-lingual Word Embeddings:	
Word Embedding Size	300
Bi-LSTM Hidden Size	300
Finetuning CharCNN Filter Number	25
Finetuning CharCNN Filter Widths	[2, 3, 4]
Optimizer	Adam [134]
Learning Rate	0.005
Batch Size	128
Cross-lingual Language Model:	
Transformer Hidden Size	512
Transformer Heads	8
Transformer Layers	3
Shared BPE units	40,000
Optimizer	Adam
Learning Rate	0.001
Batch Size	16

Table 7.2: Hyper-parameters and training setup.

7.2.3 Model Comparison

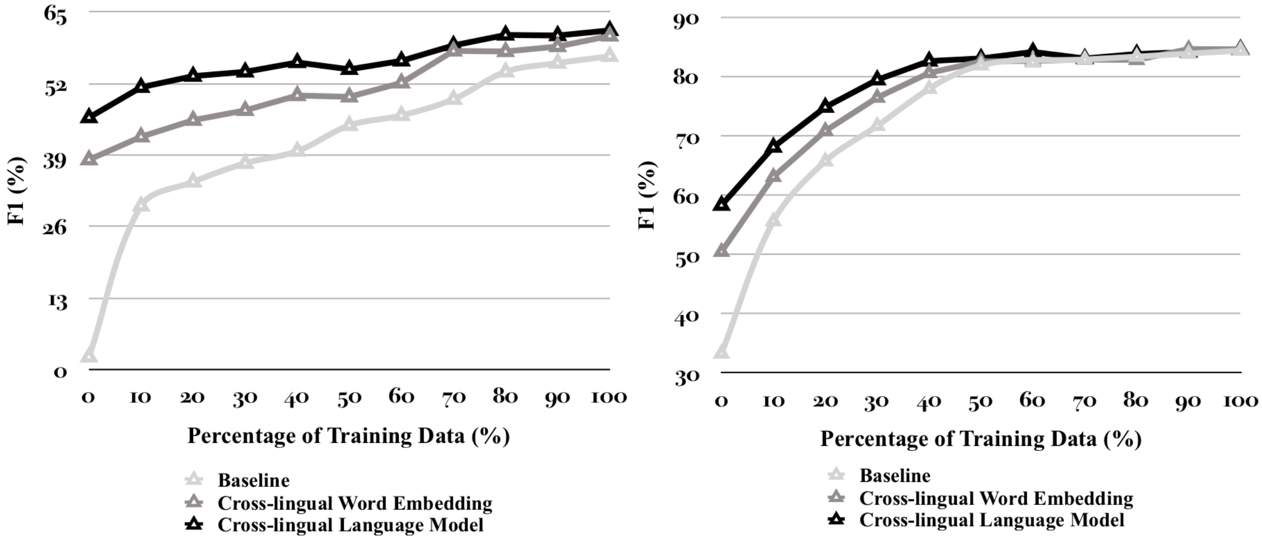
In Table 7.3, we evaluate name taggers trained with cross-lingual word embeddings and cross-lingual language model. Without any Russian/Spanish labeled data, the English name tagger trained on cross-lingual word embeddings achieved 37.81% F-1 on Russian, and 50.13% on Spanish. As capturing extra contextual information, cross-lingual language model outperforms cross-lingual word embeddings and achieves 45.79% and 58.28% on Russian and Spanish respectively.

Training Lang.	**Eval Lang.**	**Model**	**F-1 (%)**
Russian	Russian	Bi-LSTM + Char + CRF	63.77
English	Russian	Directly Transferring*	2.31
English	Russian	Cross-lingual Word Embeddings	37.81
English	Russian	Cross-lingual Language Model	**45.79**
Spanish	Spanish	Bi-LSTM + Char + CRF	83.31
English	Spanish	Directly Transferring*	33.31
English	Spanish	Cross-lingual Word Embeddings	50.13
English	Spanish	Cross-lingual Language Model	**58.28**

∗ Directly Transferring applies HL name tagger on LL sentence without any cross-lingual embeddings.

Table 7.3: Model comparisons for English-Russian and English-Spanish.

Moreover, when LL annotations are available, we initialize an LL name tagger using the parameters of the cross-lingual name tagger that is pre-trained on English annotations, and then finetune it on the LL labeled data. We hypothesize that when LL annotations are insufficient, incorporating pre-trained HL name tagger brings external information and thus improves LL name tagging performance. To investigate the impact of pre-training HL name tagger, we split LL training data into 10 portions and evaluate our method using different percentage of LL training data. 7.3a and 7.3b present two learning curves for Russian and Spanish. The finetuned models achieve 65.17% and 83.61% on Russian and Spanish, which outperforms our baselines. The Russian result is more impressive than Spanish as the Russian labeled data (4,755 sentences) is relatively smaller than Spanish labeled data (8,323). Similarly, the cross-lingual language model performs better than cross-lingual word embeddings.



(a) Russian

(b) Spanish

Figure 7.3: Learning curve of finetuning pre-trained HL name tagger on LL annotations.

Chapter 8
System Demo

8.1 System Overview

Our cross-lingual entity extraction, linking and localization system is capable of extracting named entities from unstructured text in any of 282 Wikipedia languages, translating them into English, and linking them to English Knowledge Bases (Wikipedia and Geonames). This system then produces visualizations of the results such as heatmaps, and thus it can be used by an English speaker for monitoring disasters and coordinating rescue and recovery efforts reported from incident regions in low-resource languages. In this chapter, we will present a comprehensive overview of the system components (Section 8.2 and Section 8.3), APIs (Section 8.4), interface[18](Section 8.5), and visualization[19] (Section 8.6).

APIs	Description
`/status`	Retrieve the current server status, including supported languages, language identifiers, and the state (offline, online, or pending) of each model.
`/status/{identifier}`	Retrieve the current status of a given language.
`/entity_discovery_and_linking/{identifier}`	Main entry of the EDL system. Take input in either plain text or `*.ltf` format, tag names that are PER, ORG or LOC/GPE, and link them to Wikipedia.
`/name_transliteration/{identifier}`	Transliterate a name to Latin script.
`/entity_linking/{identifier}`	Query based entity linking. Link each mention to KBs.
`/entity_linking_amr`	English entity linking for Abstract Meaning Representation (AMR) style input [135]. AMR [92] is a structured semantic representation scheme. The rich semantic knowledge in AMR boosts linking performance.
`/localize/{identifier}`	Localize a LOC/GPE name based on GeoNames database.

Table 8.1: Runtime APIs description.

8.2 Entity Extraction

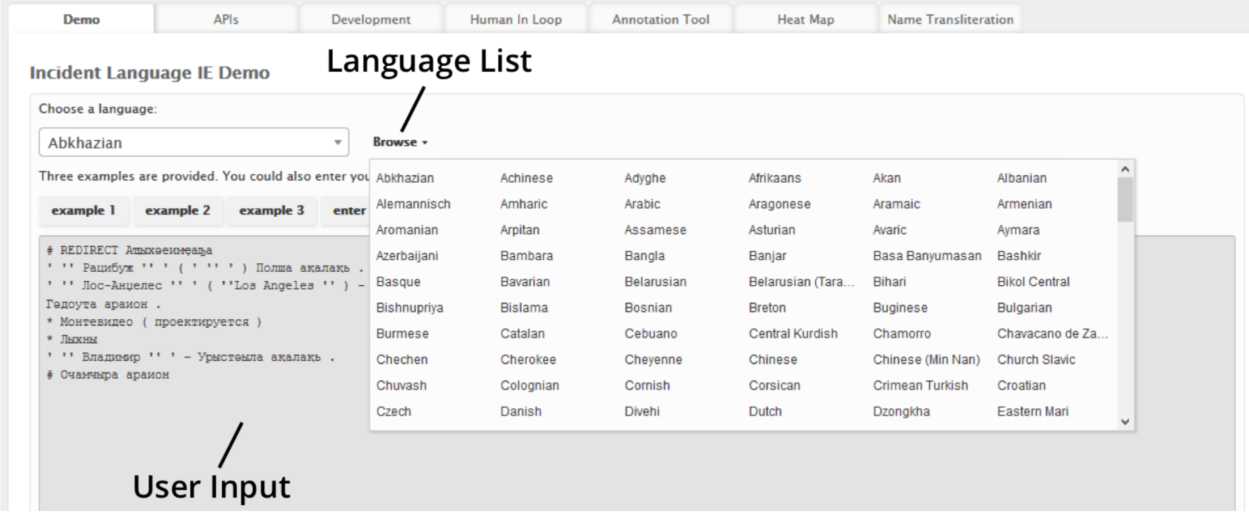
Given a text document as input, the entity extraction component identifies entity name mentions and classifies them into pre-defined types: Person (PER), Geo-political Entity

[18] `https://elisa-ie.github.io`

[19] `https://elisa-ie.github.io/heatmap`

APIs	Description
`/status`	An alias of `/status`
`/status/{identifier}`	Query the current status of a model being trained.
`/train/{identifier}`	Train a new name tagging model for a language. A model id is automatically generated and returned based on model name, and time stamp.

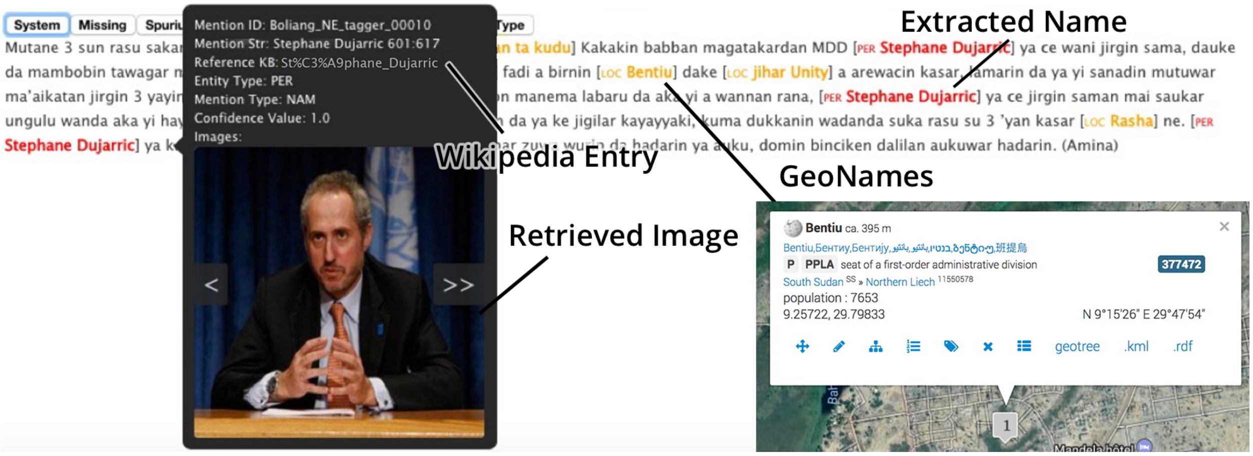
Table 8.2: Training APIs description.



Three examples are provided. You could also enter you

```
# REDIRECT Атыхәеимҿаҕа
' '' Рацибуж '' ' ( ' '' ' ) Полша аҟалакь .
' '' Лос-Анџелес '' ' ( ''Los Angeles '' ) –
Гәдоута араион .
* Монтевидео ( проектируется )
* Лыхны
' '' Владимир '' ' – Урыстәыла аҟалакь .
# Очамчыра араион
```

Figure 8.1: Cross-lingual Entity Extraction and Linking Interface



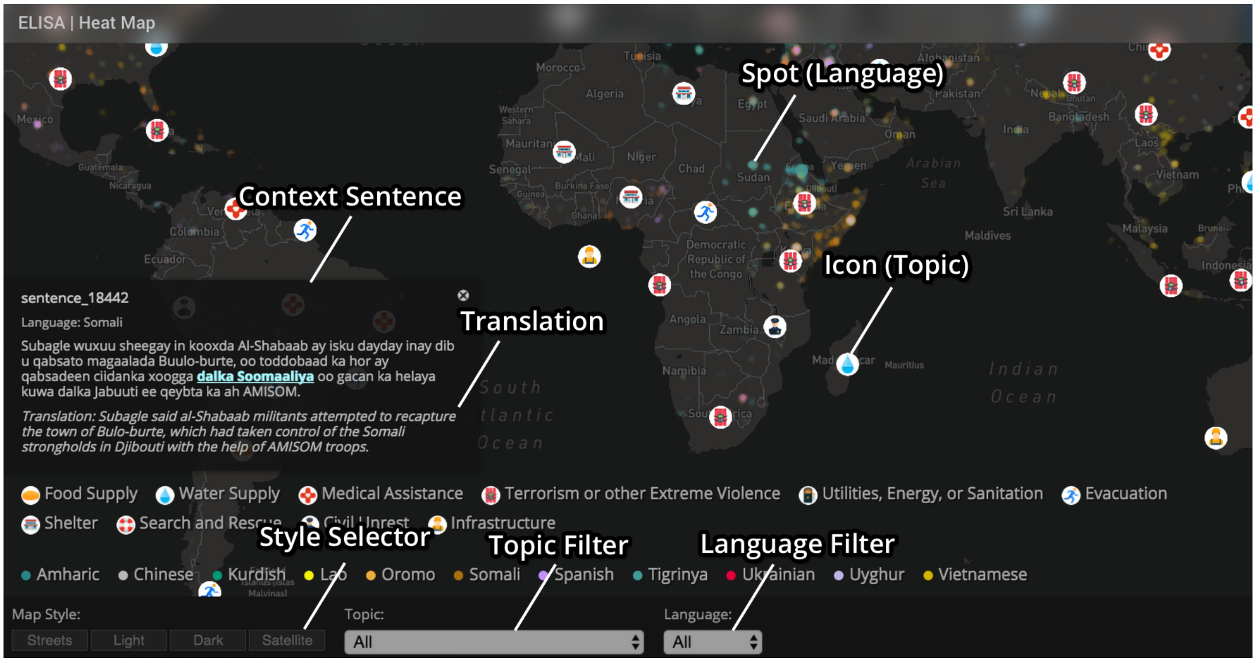
Mention ID: Boliang_NE_tagger_00010
Mention Str: Stephane Dujarric 601:617
Reference KB: St%C3%A9phane_Dujarric
Entity Type: PER
Mention Type: NAM
Confidence Value: 1.0
Images:

Mutane 3 sun rasu sakar … n ta kudu] Kakakin babban magatakardan MDD [PER Stephane Dujarric] ya ce wani jirgin sama, dauke da mambobin tawagar n … fadi a birnin [LOC Bentiu] dake [LOC jihar Unity] a arewacin kasar, hamarin da ya yi sanadin mutuwar ma'aikatan jirgin 3 yayin … n manema labaru da aka yi a wannan rana, [PER Stephane Dujarric] ya ce jirgin saman mai saukar ungulu wanda aka yi hay … da ya ke jigilar kayayyaki, kuma dukkanin wadanda suka rasu su 3 'yan kasar [LOC Rasha] ne. [PER Stephane Dujarric] ya k … ar zuwa wurin da hadarin ya auku, domin binciken dalilan aukuwar hadarin. (Amina)

Bentiu ca. 395 m
Bentiu,Бентиу,Бентију,بانتیو,بانتيو,בנטיו,ბენტიუ,班提烏
P PPLA seat of a first-order administrative division
South Sudan SS » Northern Liech 11550578
population : 7653
9.25722, 29.79833
N 9°15'26" E 29°47'54"

Figure 8.2: Cross-lingual Entity Extraction and Linking Testing Result Visualization

(GPE), Organization (ORG) and Location (LOC). We consider name tagging as a sequence labeling problem, to tag each token in a sentence as the Beginning (B), Inside (I) or Outside (O) of an entity mention with a certain type. Our model is based on a bi-directional long short-term memory (LSTM) networks with a Conditional Random Fields (CRFs) layer [102].



sentence_18442

Language: Somali

Subagle wuxuu sheegay in kooxda Al-Shabaab ay isku dayday inay dib u qabsato magaalada Buulo-burte, oo toddobaad ka hor ay qabsadeen ciidanka xoogga dalka Soomaaliya oo gacan ka helaya kuwa dalka Jabuuti ee qeybta ka ah AMISOM.

Translation: Subagle said al-Shabaab militants attempted to recapture the town of Bulo-burte, which had taken control of the Somali strongholds in Djibouti with the help of AMISOM troops.

Map Style: Streets | Light | Dark | Satellite

Topic: All

Language: All

Figure 8.3: Heatmap Visualization

It is challenging to perform entity extraction across a massive variety of languages because most languages don't have sufficient data to train a machine learning model. To tackle the low-resource challenge, we developed creative methods of deriving noisy training data from Wikipedia [85], exploiting non-traditional language-universal resources [37] and cross-lingual transfer learning [136].

8.3 Entity Linking and Localization

After we extract entity mentions, we link GPE and LOC mentions to GeoNames,[20] and PER and ORG mentions to Wikipedia.[21] We adopt the name translation approach described in [85] to translate each tagged entity mention into English, then we apply an unsupervised collective inference approach [135] to link each translated mention to the target KB. Figure 8.2 shows an example output of a Hausa document. The extracted entity mentions "*Stephane Dujarric*" and "*birnin Bentiu*" are linked to their corresponding entries in Wikipedia and GeoNames respectively.

Compared to traditional entity linking, the unique challenge of linking to GeoNames is that it is very scarce, without rich linked structures or text descriptions. Only 500k out of

[20] http://www.geonames.org
[21] https://www.wikipedia.org

Language	F1 (%)	Language	F1 (%)
Arabic	51.9	Bengali	74.8
Chechen	58.9	Persian	58.4
Hausa	70.2	Hungarian	60.2
Oromo	81.3	Russian	63.7
Somali	67.6	Tamil	65.9
Thai	69.8	Tigrinya	73.2
Tagalog	78.7	Turkish	74.4
Uyghur	72.3	Uzbek	71.8
Vietnamese	68.5	Yoruba	50.1

Table 8.3: Name Tagging Performance on Low-Resource Languages

4.7 million entities in Wikipedia are linked to GeoNames. Therefore, we associate mentions with entities in the KBs in a collective manner, based on salience, similarity and coherence measures [135]. We calculate topic-sensitive PageRank scores for 500k overlapping entities between GeoNames and Wikipedia as their salience scores. Then we construct knowledge networks from source language texts, where each node represents a entity mention, and each link represents a sentence-level co-occurrence relation. If two mentions cooccur in the same sentence, we prefer their entity candidates in the GeoNames to share an administrative code and type, or be geographically close in the world, as measured in terms of latitude and longitude.

Table 8.3 shows the performance of our system on some representative low-resource languages for which we have ground-truth annotations from the DARPA LORELEI[22] programs, prepared by the Linguistic Data Consortium.

8.4 Training and Testing APIs

In this section, we introduce our back-end APIs. The back-end is a set of RESTful APIs built with Python Flask,[23] which is a light weight framework that includes template rendering and server hosting capabilities. We use Swagger for documentation management. Besides the on-line hosted APIs, we also publish our Docker copy[24] at Dockerhub for software distribution.

In general, we categorize the APIs into two sections: RUN and TRAIN. The RUN

[22] https://www.darpa.mil/program/low-resource-languages-for-emergent-incidents

[23] http://flask.pocoo.org

[24] https://hub.docker.com/r/elisarpi/elisa-ie/

section is responsible for running the pre-trained models for 282 languages, and the TRAIN section provides a re-training function for users who want to train their own customized name tagging models using their own datasets. We also published our training and test data sets, as well as resources related to at morphology analysis and name translation at: `https://elisa-ie.github.io/wikiann`. Table 8.1 and Table 8.2 present the detailed functionality and usages of the APIs of these two sections. Besides the core components as described in Section 8.2 and Section 8.3, we also provide the APIs of additional components, including a re-trainable name transliteration component [137] and a universal name and word translation component based on word alignment derived from cross-lingual Wikipedia links [85]. More detailed usages and examples can be found in our Swagger[25] documentation: `https://elisa-ie.github.io/api`.

8.5 Testing Interface

Figure 8.1 shows the test interface, where a user can select one of the 282 languages, enter a text or select an example document, and run the system. Figure 8.2 shows an output example. In addition to the entity extraction and linking results, we also display the top 5 images for each entity retrieved from Google Image Search.[26] In this way even when a user cannot read a document in a low-resource language, s/he will obtain a high-level summary of entities involved in the document.

8.6 Heatmap Visualization

Using disaster monitoring as a use case, we detect the following ten topics from the input multi-lingual data based on translating 117 English disaster keywords via PanLex:[27] (1) water supply, (2) food supply, (3) medical assistance, (4) terrorism or other extreme violence, (5) utilities, energy or sanitation, (6) evacuation, (7) shelter, (8) search and rescue, (9) civil unrest or wide-spread crime, and (10) infrastructure, as defined in the NIST LoreHLT2017 Situation Frame detection task.[28] If a sentence includes one of these topics and also a location or geo-political entity, we will visualize the entity on a world *heatmap* using Mapbox[29] based

[25] `https://swagger.io`

[26] `https://images.google.com`

[27] `http://panlex.org`

[28] `https://www.nist.gov/itl/iad/mig/lorehlt-evaluations`

[29] `https://www.mapbox.com`

on its coordinates in the GeoNames database obtained from the entity linker. We also show the entire context sentence and its English translation produced from our state-of-the-art Machine Translation system for low-resource languages [136]. Figure 8.3 illustrates an example of the visualized heatmap.

We use different colors and icons to stand for different languages and frame topics respectively (e.g., the bread icon represents “food supply”). Users can also specify the language or frame topic or both to filter out irrelevant results on the map. By clicking an icon, its context sentence will be displayed in a pop-up with automatic translation and highlighted mentions and keywords. We provide various map styles (light, dark, satellite, and streets) for different needs, as shown in Figure 8.4.

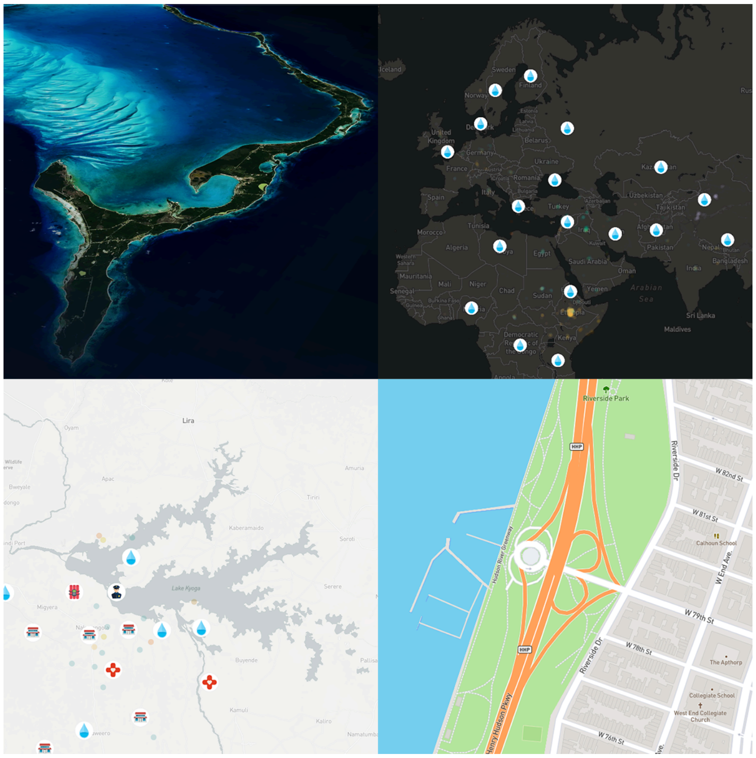


Figure 8.4: Different Map Styles

Chapter 9
Conclusions and Future Work

9.1 Conclusions

In this thesis, we first presented readers an overview of information extraction and name tagging, as well as a brief introduction of low-resource languages (LL). To address the challenges posed by name tagging for low-resource languages, we formulated four hypotheses and propose concrete solutions to justify each hypothesis. At the end, we provided a system demo that makes our research outcomes publicly available.

In Chapter 2, we walked through the history of name tagging, which dates back to 1995. At the time, Message Understanding Conference (MUC-6) added the task of recognition of named entities, and since then, name tagging became an active area of research for past twenty years. We started from introducing traditional name tagging approaches, such as rule-based systems, Hidden Markov Model (HMM), Support Vector Machine (SVM) and Conditional Random Field (CRF). Then we presented neural network based approaches that achieved remarkable performance in recent years. At last, we showed the bottleneck of these methods to low-resource language name tagging.

In Chapter 3, we presented solutions to our first hypothesis which assumes that noisy low-resource language annotations created by non-native speakers or automatically distilled from existing resources can provide weak supervision to machine learning models. We introduced 1) "Chinese Room" which is a annotation platform for non-name speakers, and 2) name projection through parallel data.

In Chapter 4 and Chapter 5, we justified our second hypothesis: language universal features can mitigate the impact of noise in annotations, and provide robustness and generalization to a weakly supervised machine learning model. In situations where LL labeled data are insufficient and noisy, traditional machine learning methods suffer from huge performance decreases. We proposed a neural framework that can incorporates many non-traditional language universal resources that are readily available but rarely explored in the NLP community. We encoded such various types of non-traditional linguistic resources as features into a supervised DNN name tagger. Our proposed architecture significantly outperform baseline methods.

In Chapter 6, we observed that by only relying on local contextual information, the current DNN models perform poorly when the local context is ambiguous or limited. To address this problem, we formulated the third hypothesis: recognizing names not only relies on the context of the sentence, but also the context of the article, sometime even the whole corpus. We proposed a new framework to improve the DNN name tagger by utilizing local and global (document-level and corpus-level) contextual information. We retrieved the document-level context from other sentences within the same document and corpus-level context from sentences in other documents. The proposed model learns to incorporate document-level and corpus-level contextual information alongside local contextual information via global attention, which dynamically weights their respective contextual information, and gating mechanisms, which determine the influence of this information.

In Chapter 7, we investigated the problem of training LL name taggers without using any LL labeled data. Our proposed approaches justified the forth hypothesis that distributional word embeddings/contextualized word embeddings of HL and LL can be projected into a shared space, so that machine learning models trained on HL embeddings/contextualized embeddings can produce satisfying performance on LL. We discussed the advantages of pre-trained language model over word embeddings. In experiments, we use MUSE which a word embedding alignment tool [26] to generate cross-lingual word embeddings. And we proposed a new architecture to pre-train cross-lingual language models. At last, we evaluated cross-lingual name taggers trained with cross-lingual word embeddings and cross-lingual language model, and showed that cross-lingual language model achieves better performance on name tagging task.

In Chapter 8, we fused our research outcomes together and introduced a publicly available system that is capable of extracting named entities from unstructured text in any of 282 Wikipedia languages, translating them into English, and linking them to English Knowledge Bases (Wikipedia and Geonames). The APIs have been used by many third party organizations.

All together, we are really excited about the progress that has been made in this field in recent years and have been glad to be able to contribute to this field.

9.2 Future Work

We believe that there is still a long way to go towards fully unsupervised low-resource name tagging, and we are still facing enormous challenges and a lot of questions that we need to address in the future. Here we present several research directions and some potential solutions from our perspective.

9.2.1 Enhanced Cross-lingual Contextualized Word Embedding

In the thesis, we show that cross-lingual contextualized word embedding significantly improves language transferring capability of name tagging, due to its awareness of context information. However, there is still a gap between the cross-lingual name tagger and the name tagger that is trained on low-resource language annotations. We show some limitations of the current method, and to bridge the gap, we point out some future directions.

- The first limitation is "word order shift" where words in phrases of two languages are in different order due to inconsistent syntax, e.g. "*University of Washington*" is "华盛顿*(Washington)* 大学*(University)*" in Chinese. To address this problem, other than the method we used in previous experiments where we removed B and I tags from BIO schema, we propose to train cross-lingual contextualized word embedding based on span representation instead of word representation. By doing this, we change from training a model for word-for-word translation to phrase-for-phrase translation. [138, 139] show span representations are effective for many tasks.

- Training contextualized word embedding is computationally expensive. [30, 29, 27] compare models with different sizes of parameters, from a Bi-LSMT encoder of 512 dimension to multi-layer transformer encoder with 340 million parameters. Larger size of parameters significantly improve representation quality and yield better performance on various NLP tasks. Therefore, it is important to optimize the space complexity and time complexity when training neural models so that we can apply larger models and obtain better representations. Potential optimization methods include: 1) changing computation precision from floating point 32 (default in most deep learning frameworks) to floating point 16 to save GPU memory and speed up training, 2) replace all general softmax layer with hierarchical softmax [140], and 3) negetive sample [141] to reduce candidate sizes.

- Current method to train cross-lingual contextualized word embedding is purely unsupervised and cannot utilize human annotations when available. [86] shows that adding a small amount of supervision into an unsupervised model significantly improves model performance. We propose to use parallel data to finetune an unsupervisedly pre-trained contextualized word embedding. [142] shares similar idea with us but uses a different method.

With better quality cross-lingual contextualized word embedding, a name tagger trained on HL annotations can better understand LL and thus achieves improved performance.

9.2.2 Unsupervised Cross-lingual Information Extraction

Training cross-lingual contextualized word embedding essentially learns how to translate words from one language to another. It is not limited to the task of name tagging, but can also be applied to many other cross-lingual NLP tasks, such as event extraction and relation extraction. With cross-lingual contextualized word embedding, we can build a comprehensive low-resource language information extraction framework by transferring knowledge from HL to LL. It only needs existing HL training data and requires no LL annotations.

Event Extraction is one of the core Information Extraction (IE) tasks that aims to identify event triggers and arguments from unstructured texts and classify them into predefined categories. For example, given a sentence “Tim Cook joined Apple in 1998.”, a event extraction system should discover 1) “joined” as an **event trigger** which is a word or phrase that clearly presents the occurrence of an event, 2) “Tim Cook” and “Apple” as PERSON and ORGANIZATION **event arguments** of the event trigger “joined”, and 3) assign the whole sentence a pre-defined **event type** Personnel.Start-POSITION. Event trigger and event argument labeling are usually considered as sequence tagging problem, which is similar to name tagging. We can use cross-lingual contextualized word embedding in the same way as name tagging to build an LL event trigger and argument tagger from HL resources. Event type prediction is usually considered as a multi-label classification problem, where the input to the classifier is word embedding, followed by convolutional networks to extract features and predict labels. We can simply replace the word embedding by cross-lingual word embedding or contextualized embedding during training, and obtain a cross-lingual event type classifier.

As what we mentioned so far, to grant cross-lingual capability to a machine learning model, we replace the word embedding in neural models with cross-lingual contextualized word embedding and train it on HL annotations, and then we are able to directly evaluate them on LL. This procedure is applicable to relation extraction as well. Relation extraction aims to predict a pre-defined relation type between two given entity mentions for a sentence. For example, in "**Steve Jobs** is the co-founder of **Apple**.", the pre-defined relation between entity mentions "**Steve Jobs**" and "**Apple**" is *Org-Affiliation-Founder*. A typical approach for relation extraction is CNN based supervised classifier. We similarly replace the word embedding of the model by cross-lingual contextualized embedding during training, and train the cross-lingual relation classifier on HL annotations.

9.2.3 Joint Information Extraction

In the realm of information extraction, inter-dependencies and constraints across multiple tasks and multiple languages are pervasive. One task can benefit from interactions among multiple relevant tasks, such as name tagging and entity linking. Entity linking is a task of linking a name mention to a unique entry in a Knowledge Base (KB). A joint approach of name tagging and entity linking can significantly eliminate ambiguities for both tasks, for example, if a tagged phrase has no entry in KB, then it's not likely to be a name, and if a name is a person, then it's likely to be linked to a person entry in KB.

Joint information extraction has been extensively studied in traditional feature based machine learning approaches. However, in the neural network "era", although there are several attempts that have been made to jointly learning across multiple tasks, we believe that the way to jointly model multiple tasks and consider all possible inter-dependencies is still far from perfect.

9.3 Final Remark

As the final remark, this thesis, by standing upon the shoulders of previous research, studied the topic of name tagging for low-resource languages, and provided a different angle to contribute to the task. We hope the observations, hypotheses, approaches and experiment results of this thesis are able to inspire further research in the field of name tagging or other related areas.

REFERENCES

Bibliography

[1] J. Searle, “Minds, brains, and programs,” *Journal of the Association for Computing Machinery*, 1980.

[2] D. Jurafsky, *Speech & language processing.* Pearson Education India, 2000.

[3] H. Ji, R. Grishman, and H. T. Dang, “Overview of the tac 2010 knowledge base population track,” 2010.

[4] H. Ji and R. Grishman, “Knowledge base population: Successful approaches and challenges,” in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1.* Association for Computational Linguistics, 2011, pp. 1148–1158.

[5] X. Yao and B. Van Durme, “Information extraction over structured data: Question answering with freebase,” in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2014, pp. 956–966.

[6] R. Srihari and W. Li, “Information extraction supported question answering,” CYMFONY NET INC WILLIAMSVILLE NY, Tech. Rep., 1999.

[7] J. L. Leidner, G. Sinclair, and B. Webber, “Grounding spatial named entities for information extraction and question answering,” in *Proceedings of the HLT-NAACL 2003 workshop on Analysis of geographic references-Volume 1.* Association for Computational Linguistics, 2003, pp. 31–38.

[8] Y. Chen, C. Zong, and K.-Y. Su, “A joint model to identify and align bilingual named entities,” *Computational Linguistics*, vol. 39, no. 2, pp. 229–266, 2013.

[9] D. Mollá, M. Van Zaanen, D. Smith *et al.*, “Named entity recognition for question answering,” 2006.

[10] Á. Rodrigo, J. Pérez-Iglesias, A. Peñas, G. Garrido, and L. Araujo, “Answering questions about european legislation,” *Expert Systems with Applications*, vol. 40, no. 15, pp. 5811–5816, 2013.

[11] C. Cieri, M. Maxwell, S. M. Strassel, J. Tracey, K. Choukri, T. Declerck, S. Goggi, M. Grobelnik *et al.*, “Selection criteria for low resource language programs.” in *LREC*, 2016.

[12] K. Megerdoomian and D. Parvaz, “Low-density language bootstrapping: the case of tajiki persian.” in *LREC*, 2008.

[13] C. Hogan, “Ocr for minority languages,” in *Symposium on Document Image Understanding Technology*, 1999.

[14] F. J. Och and H. Ney, "A systematic comparison of various statistical alignment models," *Computational linguistics*, 2003.

[15] J. Mortimer and J. Salathiel, "'soundex'codes of surnames provide confidentiality and accuracy in a national hiv database." *Communicable disease report. CDR review*, 1995.

[16] H. Raghavan and J. Allan, "Using soundex codes for indexing names in asr documents," in *Proceedings of the Workshop on Interdisciplinary Approaches to Speech Indexing and Retrieval*, 2004.

[17] X. Pan, B. Zhang, J. May, J. Nothman, K. Knight, and H. Ji, "Cross-lingual name tagging and linking for 282 languages," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2017, pp. 1946–1958.

[18] M. S. Dryer and M. Haspelmath, "The world atlas of language structures online," in *Leipzig: Max Planck Institute for Evolutionary Anthropology*, 2013.

[19] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *Proc. ACL1995*, 2003.

[20] R. Florian, H. Hassan, A. Ittycheriah, H. Jing, N. Kambhatla, X. Luo, N. Nicolov, and S. Roukos, "A statistical model for multilingual entity detection and tracking," in *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL 2004)*, 2004.

[21] U. Hermjakob, Q. Li, D. Marcu, J. May, S. J. Mielke, N. Pourdamghani, M. Pust, X. Shi, K. Knight, T. Levinboim, K. Murray, D. Chiang, B. Zhang, X. Pan, D. Lu, Y. Lin, and H. Ji, "Incident-driven machine translation and name tagging for low-resource languages," *Machine Translation*, pp. 1–31, 2017.

[22] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[23] M. Faruqui and C. Dyer, "Improving vector space word representations using multilingual correlation," in *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, 2014, pp. 462–471.

[24] C. Xing, D. Wang, C. Liu, and Y. Lin, "Normalized word embedding and orthogonal transform for bilingual word translation," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 1006–1011.

[25] S. L. Smith, D. H. Turban, S. Hamblin, and N. Y. Hammerla, "Offline bilingual word vectors, orthogonal transformations and the inverted softmax," *arXiv preprint arXiv:1702.03859*, 2017.

[26] A. Conneau, G. Lample, M. Ranzato, L. Denoyer, and H. Jégou, "Word translation without parallel data," *arXiv preprint arXiv:1710.04087*, 2017.

[27] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” *arXiv preprint arXiv:1810.04805*, 2018.

[28] A. M. Dai and Q. V. Le, “Semi-supervised sequence learning,” in *Advances in neural information processing systems*, 2015, pp. 3079–3087.

[29] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, “Deep contextualized word representations,” *arXiv preprint arXiv:1802.05365*, 2018.

[30] M. E. Peters, W. Ammar, C. Bhagavatula, and R. Power, “Semi-supervised sequence tagging with bidirectional language models,” *arXiv preprint arXiv:1705.00108*, 2017.

[31] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, “Improving language understanding by generative pre-training,” *URL https://s3-us-west-2. amazonaws. com/openai-assets/research-covers/languageunsupervised/language understanding paper. pdf*, 2018.

[32] J. Howard and S. Ruder, “Universal language model fine-tuning for text classification,” *arXiv preprint arXiv:1801.06146*, 2018.

[33] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[34] G. Lample, A. Conneau, L. Denoyer, and M. Ranzato, “Unsupervised machine translation using monolingual corpora only,” *arXiv preprint arXiv:1711.00043*, 2017.

[35] J. P. Chiu and E. Nichols, “Named entity recognition with bidirectional lstm-cnns,” *Transactions of the Association for Computational Linguistics*, vol. 4, pp. 357–370, 2016.

[36] N. Chinchor, “Muc-4 evaluation metrics,” in *Proceedings of the 4th conference on Message understanding.* Association for Computational Linguistics, 1992, pp. 22–29.

[37] B. Zhang, X. Pan, T. Wang, A. Vaswani, H. Ji, K. Knight, and D. Marcu, “Name tagging for low-resource incident languages based on expectation-driven learning,” in *Proceeddings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies*, 2016.

[38] B. Zhang, D. Lu, X. Pan, Y. Lin, H. Abudukelimu, H. Ji, and K. Knight, “Embracing non-traditional linguistic resources for low-resource language name tagging,” in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2017.

[39] B. Zhang, Y. Lin, X. Pan, D. Lu, J. May, K. Knight, and H. Ji, “Elisa-edl: A cross-lingual entity extraction, linking and localization system,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, 2018, pp. 41–45.

[40] D. Farmakiotou, V. Karkaletsis, J. Koutsias, G. Sigletos, C. D. Spyropoulos, and P. Stamatopoulos, "Rule-based named entity recognition for greek financial texts," in *Proceedings of the Workshop on Computational lexicography and Multimedia Dictionaries (COMLEX 2000)*, 2000.

[41] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Lingvisticae Investigationes*, 2007.

[42] V. Yadav and S. Bethard, "A survey on recent advances in named entity recognition from deep learning models," in *Proceedings of the 27th International Conference on Computational Linguistics*, 2018, pp. 2145–2158.

[43] D. M. Bikel, S. Miller, R. Schwartz, and R. Weischedel, "Nymble: a high-performance learning name-finder," in *Proceedings of the fifth conference on Applied natural language processing.* Association for Computational Linguistics, 1997.

[44] A. McCallum, D. Freitag, and F. C. Pereira, "Maximum entropy markov models for information extraction and segmentation." in *Icml*, vol. 17, no. 2000, 2000, pp. 591–598.

[45] G. Zhou and J. Su, "Named entity recognition using an hmm-based chunk tagger," in *proceedings of the 40th Annual Meeting on Association for Computational Linguistics.* Association for Computational Linguistics, 2002, pp. 473–480.

[46] K. Seymore, A. McCallum, and R. Rosenfeld, "Learning hidden markov model structure for information extraction," in *Proc. AAAI-99 workshop on machine learning for information extraction*, 1999.

[47] M. Asahara and Y. Matsumoto, "Japanese named entity extraction with redundant morphological analysis," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1.* Association for Computational Linguistics, 2003, pp. 8–15.

[48] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[49] A. McCallum and W. Li, "Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons," in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4.* Association for Computational Linguistics, 2003, pp. 188–191.

[50] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[51] Y. Li, K. Bontcheva, and H. Cunningham, "Svm based learning system for information extraction," in *International Workshop on Deterministic and Statistical Methods in Machine Learning.* Springer, 2004, pp. 319–339.

[52] I. Segura-Bedmar, P. Martínez, and M. H. Zazo, "Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013)," in *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, vol. 2, 2013, pp. 341–350.

[53] S. Liu, B. Tang, Q. Chen, and X. Wang, "Effects of semantic features on machine learning-based drug name recognition systems: word embeddings vs. manually constructed dictionaries," *Information*, vol. 6, no. 4, pp. 848–865, 2015.

[54] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the 25th international conference on Machine learning.* ACM, 2008, pp. 160–167.

[55] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *Journal of machine learning research*, vol. 12, no. Aug, pp. 2493–2537, 2011.

[56] T. Mikolov, W.-t. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2013, pp. 746–751.

[57] A. Graves, N. Jaitly, and A.-r. Mohamed, "Hybrid speech recognition with deep bidirectional lstm," in *Automatic Speech Recognition and Understanding, 2013 IEEE Workshop on*, 2013.

[58] Z. Huang, W. Xu, and K. Yu, "Bidirectional lstm-crf models for sequence tagging," *arXiv preprint arXiv:1508.01991*, 2015.

[59] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, "Neural architectures for named entity recognition," in *Proceedings of 2016 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2016.

[60] X. Ma and E. Hovy, "End-to-end sequence labeling via bi-directional lstm-cnns-crf," *arXiv preprint arXiv:1603.01354*, 2016.

[61] L. Liu, J. Shang, F. Xu, X. Ren, H. Gui, J. Peng, and J. Han, "Empower sequence labeling with task-aware neural language model," 2017.

[62] M. Sato, H. Shindo, I. Yamada, and Y. Matsumoto, "Segment-level neural conditional random fields for named entity recognition," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing*, 2017.

[63] Z. Yang, R. Salakhutdinov, and W. W. Cohen, "Transfer learning for sequence tagging with hierarchical recurrent networks," *arXiv preprint arXiv:1703.06345*, 2017.

[64] Q. Li, H. Li, H. Ji, W. Wang, J. Zheng, and F. Huang, “Joint bilingual name tagging for parallel corpora,” in *Proceedings of the 21st ACM international conference on Information and knowledge management*, 2012.

[65] S. Kim, K. Toutanova, and H. Yu, “Multilingual named entity recognition using parallel data and metadata from wikipedia,” in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, 2012.

[66] W. Che, M. Wang, C. D. Manning, and T. Liu, “Named entity recognition with bilingual constraints.” in *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics*, 2013.

[67] M. Wang, W. Che, and C. D. Manning, “Joint word alignment and bilingual named entity recognition using dual decomposition.” in *Proceedings of the Association for Computational Linguistics*, 2013.

[68] J. Nothman, N. Ringland, W. Radford, T. Murphy, and J. R. Curran, “Learning multilingual named entity recognition from wikipedia,” *Artificial Intelligence*, 2013.

[69] A. Mikheev, M. Moens, and C. Grover, “Named entity recognition without gazetteers,” in *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*, 1999.

[70] O. Etzioni, M. Cafarella, D. Downey, A.-M. Popescu, T. Shaked, S. Soderland, D. S. Weld, and A. Yates, “Unsupervised named-entity extraction from the web: An experimental study,” *Artificial intelligence*, 2005.

[71] D. Nadeau, P. Turney, and S. Matwin, “Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity,” 2006.

[72] H. Ji and D. Lin, “Gender and animacy knowledge discovery from web-scale n-grams for unsupervised person mention detection,” in *Proceedings of PACLIC2009*, 2009.

[73] J. An, S. Lee, and G. G. Lee, “Automatic acquisition of named entity tagged corpus from world wide web,” in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, 2003.

[74] M. Mintz, S. Bills, R. Snow, and D. Jurafsky, “Distant supervision for relation extraction without labeled data,” in *Proceeddings of the conference of the Association for Computational Linguistics*, 2009.

[75] X. Ren, A. El-Kishky, C. Wang, F. Tao, C. R. Voss, H. Ji, and J. Han, “Clustype: Effective entity recognition and typing by relation phrase-based clustering,” in *Proceeddings of the 21st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2015.

[76] Q. Li, H. Li, H. Ji, W. Wang, J. Zheng, and F. Huang, “Joint bilingual name tagging for parallel corpora,” in *Proceedings of The Conference on Information and Knowledge Management*, 2012.

[77] M. Wang and C. Manning, “Cross-lingual projected expectation regularization for weakly supervised learning,” in *Transactions of the Association of Computational Linguistics*, 2014.

[78] D. Zhang, B. Zhang, X. Pan, and H. Ji, “Bitext name tagging for annotation projection,” in *Proceedings of the 26th International Conference on Computational Linguistics*, 2016.

[79] J. Nothman, J. R. Curran, and T. Murphy, “Transforming wikipedia into named entity training data,” in *Proceedings of the Australasian Language Technology Association Workshop 2008*, 2008.

[80] W. Dakka and S. Cucerzan, “Augmenting wikipedia with named entity tags,” in *Proceedings of the International Joint Conference on Natural Language Processing*, 2008.

[81] N. Ringland, J. Nothman, T. Murphy, and J. R. Curran, “Classifying articles in english and german wikipedia,” in *Proceedings of Australasian Language Technology Association Workshop 2009*, 2009.

[82] F. Alotaibi and M. Lee, “Mapping arabic wikipedia into the named entities taxonomy,” in *Proceedings of the International Conference on Computational Linguistics*, 2012.

[83] J. Nothman, N. Ringland, W. Radford, T. Murphy, and J. R. Curran, “Learning multilingual named entity recognition from Wikipedia,” *Artificial Intelligence*, 2012.

[84] M. Althobaiti, U. Kruschwitz, and M. Poesio, “Automatic creation of arabic named entity annotated corpus using wikipedia,” in *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, 2014.

[85] X. Pan, B. Zhang, J. May, J. Nothman, K. Knight, and H. Ji, “Cross-lingual name tagging and linking for 282 languages,” in *Proc. the 55th Annual Meeting of the Association for Computational Linguistics*, 2017.

[86] Y. Lin, S. Yang, V. Stoyanov, and H. Ji, “A multi-lingual multi-task architecture for low-resource sequence labeling,” in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 799–809.

[87] C. D. Manning, M. Surdeanu, J. Bauer, J. Finkel, S. J. Bethard, and D. McClosky, “The stanford corenlp natural language processing toolkit,” in *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics*, 2014.

[88] M. K. Odell, *The Profit in Records Management.* Systems (New York), 1956.

[89] L. Philips, “Hanging on the metaphone,” *Computer Language*, vol. 7, no. 12, 1990.

[90] R. L. Taft, *Name Search Techniques.* Albany, New York, US: New York State Identification and Intelligence System, 1970.

[91] F. J. Och and H. Ney, “A systematic comparison of various statistical alignment models,” *Computational Linguistics*, 2003.

[92] L. Banarescu, C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer, and N. Schneider, “Abstract meaning representation for sembanking,” in *ACL Workshop on Linguistic Annotation and Interoperability with Discourse*, 2013.

[93] E. Agichtein and L. Gravano, “Snowball: Extracting relations from large plain-text collections,” in *Proceedings of the fifth ACM conference on Digital libraries*, 2000.

[94] M. Collins and Y. Singer, “Unsupervised models for named entity classification,” in *Proceedings of the joint SIGDAT conference on empirical methods in natural language processing and very large corpora*, 1999.

[95] Z. Kozareva, “Bootstrapping named entity recognition with automatically generated gazetteer lists,” in *Student Research Workshop*, 2006.

[96] E. Riloff and J. Wiebe, “Learning extraction patterns for subjective expressions,” in *Proceedings of the 2003 conference on Empirical methods in natural language processing*, 2003.

[97] C. Niu, W. Li, J. Ding, and R. K. Srihari, “A bootstrapping approach to named entity classification using successive learners,” in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics, 2003, pp. 335–342.

[98] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, “Conditional random fields: Probabilistic models for segmenting and labeling sequence data,” in *Proceedings of the Eighteenth International Conference on Machine Learning*, 2001.

[99] B. Settles, “Active learning literature survey,” *University of Wisconsin, Madison*, 2010.

[100] F. Jing, M. Li, H. Zhang, and B. Zhang, “Entropy-based active learning with support vector machines for content-based image retrieval,” in *Proceedings of ICMCS2004*, 2004.

[101] B. Settles and M. Craven, “An analysis of active learning strategies for sequence labeling tasks,” in *Proceedings of the conference on empirical methods in natural language processing*, 2008.

[102] J. P. Chiu and E. Nichols, “Named entity recognition with bidirectional lstm-cnns,” in *Transaction of Association for Computational Linguistics*, 2016.

[103] G. Lample, M. Ballesteros, K. Kawakami, S. Subramanian, and C. Dyer, “Neural architectures for named entity recognition,” in *Proceeddings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies*, 2016.

[104] D. Zeng, K. Liu, S. Lai, G. Zhou, and J. Zhao, "Relation classification via convolutional deep neural network," in *Proceeddings of the 25th International Conference on Computational Linguistics*, 2014.

[105] Y. Liu, F. Wei, S. Li, H. Ji, and M. Zhou, "A dependency-based neural network for relation classification," in *Proceeddings of the 53rd Annual Meeting of the Association for Computational Linguistics*, 2015.

[106] T. H. Nguyen and R. Grishman, "Relation extraction: Perspective from convolutional neural networks," in *Proceedings of NAACL Workshop on Vector Space Modeling for NLP*, 2015.

[107] Y. Yang, Y. Tong, S. Ma, and Z.-H. Deng, "A position encoding convolutional neural network based on dependency tree for relation classification," in *Proceedings of the Empirical Methods on Natural Language Processing*, 2016.

[108] Y. Chen, L. Xu, K. Liu, D. Zeng, and J. Zhao, "Event extraction via dynamic multi-pooling convolutional 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*, 2015.

[109] T. H. Nguyen and R. Grishman, "Event detection and domain adaptation with convolutional 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*, 2015.

[110] ——, "Joint event extraction via recurrent neural networks," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, 2016.

[111] X. Feng, H. Ji, D. Tang, B. Qin, and T. Liu, "A language-independent neural network for event detection," in *Proceeddings of the 54th Annual Meeting of the Association for Computational Linguistics*, 2016.

[112] S. K. Sarma, D. Sarmah, B. Brahma, M. Mahanta, H. Bharali, and U. Saikia, "Building multilingual lexical resources using wordnets: Structure, design and implementation," in *Proceedings of the 3rd Workshop on Cognitive Aspects of the Lexicon*, 2012.

[113] R. Ostling, "Word order typology through multilingual word alignment," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*, 2015.

[114] M. S. Dryer and M. Haspelmath, Eds., *WALS Online*, 2013.

[115] T. Baldwin, J. Pool, and S. Colowick, "Panlex and lextract: Translating all words of all languages of the world," in *Proceedings of the 23rd International Conference on Computational Linguistics*, 2010.

[116] D. Kamholz, J. Pool, and S. Colowick, "Panlex: Building a resource for panlingual lexical translation," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation*, 2014.

[117] F. Bond and K. Paik, "A survey of wordnets and their licenses," in *Proceedings of the 6th Global WordNet Conference*, 2012.

[118] G. de Melo and G. Weikum, "Towards a universal wordnet by learning from combined evidence," in *Proceeddings of The Conference on Information and Knowledge Management*, 2019.

[119] ——, "Towards universal multilingual knowledge bases," in *Proceedings of the 5th Global Wordnet Conference*, 2010.

[120] G. de Melo, "Etymological wordnet: Tracing the history of words," in *Proceeddings of the Conference on Language Resources*, 2014.

[121] R. Steinberger, B. Pouliquen, M. Kabadjov, and E. van der Goot, "Jrc-names: A freely available, highly multilingual named entity resource," in *Proceeddings of the 8th International Conference on Recent Advances in Natural Language Processing*, 20011.

[122] W. A. Gale, K. W. Church, and D. Yarowsky, "One sense per discourse," in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 233–237.

[123] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proceedings of the 2015 International Conference on Learning Representations*, 2015.

[124] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, 2003.

[125] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the conll-2003 shared task: Language-independent named entity recognition," in *Proceedings of the 2003 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2003.

[126] N. Reimers and I. Gurevych, "Optimal hyperparameters for deep lstm-networks for sequence labeling tasks," *arXiv preprint arXiv:1707.06799*, 2017.

[127] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proceedings of the 2010 International Conference on Computational Statistics*, 2010.

[128] D. Gillick, C. Brunk, O. Vinyals, and A. Subramanya, "Multilingual language processing from bytes," *arXiv preprint arXiv:1512.00103*, 2015.

[129] G. Luo, X. Huang, C.-Y. Lin, and Z. Nie, "Joint entity recognition and disambiguation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015.

[130] C. Chelba, T. Mikolov, M. Schuster, Q. Ge, T. Brants, P. Koehn, and T. Robinson, "One billion word benchmark for measuring progress in statistical language modeling," *arXiv preprint arXiv:1312.3005*, 2013.

[131] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, “Enriching word vectors with subword information,” *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[132] G. Lample, M. Ott, A. Conneau, L. Denoyer, and M. Ranzato, “Phrase-based & neural unsupervised machine translation,” *arXiv preprint arXiv:1804.07755*, 2018.

[133] R. Sennrich, B. Haddow, and A. Birch, “Neural machine translation of rare words with subword units,” *arXiv preprint arXiv:1508.07909*, 2015.

[134] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[135] X. Pan, T. Cassidy, U. Hermjakob, H. Ji, and K. Knight, “Unsupervised entity linking with abstract meaning representation,” in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies*, 2015.

[136] L. Cheung, T. Gowda, U. Hermjakob, N. Liu, J. May, A. Mayn, N. Pourdamghani, M. Pust, K. Knight, N. Malandrakis, P. Papadopoulos, A. Ramakrishna, K. Singla, V. Martinez, C. Vaz, D. Can, S. Narayanan, K. Murray, T. Nguyen, D. Chiang, X. Pan, B. Zhang, Y. Lin, D. Lu, L. Huang, K. Blissett, T. Zhang, H. Ji, O. Glembek, M. K. Baskar, S. Kesiraju, L. Burget, K. Benes, I. Szoke, K. Vesely, J. H. Cernocky, C. Goudeseune, M. H. Johnson, L. Sari, W. Chen, and A. Liu, “ELISA system description for lorehlt 2017,” in *Proc. LoReHLT2017*, 2017.

[137] Y. Lin, X. Pan, A. Deri, H. Ji, and K. Knight, “Leveraging entity linking and related language projection to improve name transliteration,” in *Proc. ACL2016 Workshop on Named Entities*, 2016.

[138] K. Lee, S. Salant, T. Kwiatkowski, A. Parikh, D. Das, and J. Berant, “Learning recurrent span representations for extractive question answering,” *arXiv preprint arXiv:1611.01436*, 2016.

[139] K. Lee, L. He, M. Lewis, and L. Zettlemoyer, “End-to-end neural coreference resolution,” *arXiv preprint arXiv:1707.07045*, 2017.

[140] E. Grave, A. Joulin, M. Cissé, H. Jégou *et al.*, “Efficient softmax approximation for gpus,” in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 1302–1310.

[141] Y. Goldberg and O. Levy, “word2vec explained: deriving mikolov et al.’s negative-sampling word-embedding method,” *arXiv preprint arXiv:1402.3722*, 2014.

[142] G. Lample and A. Conneau, “Cross-lingual language model pretraining,” *CoRR*, 2019.