

Department of
Informatics, Systems and Communication

Dottorato di Ricerca in / PhD program Informatica Ciclo / Cycle XXIX

Towards Adaptation of Named Entity
Extraction and Linking Frameworks

Cognome / Surname Manchanda Nome / Name Pikakshi

Matricola / Registration number 787961

Tutore / Tutor: Giancarlo Mauri

Cotutore / Co-tutor: Dr. Matteo Palmonari

Supervisor: Dr. Elisabetta Fersini

Coordinatore / Coordinator: Prof. Stefania Bandini

ACADEMIC YEAR 2015/16

Dedication

For my Mom...

The one person who has always stood with me through thick and thin, has been a beacon of hope and a source of inspiration throughout my life. I love you, Mother. Together we can!

Abstract

English. Natural Language Processing and Knowledge Base Experts are actively involved in extracting structured information from the Unstructured Web in order to realize the Semantic Web Vision. Diverse forms of unstructured information is easily available today to research scientists from social media platforms such as Twitter and Facebook in real time. The comprehensive and widespread use of such platforms in the modern age has led to a continuous stream of evolving information along with a constant presence of noise, and ambiguity that makes the task of extracting structured information difficult. An essential step is therefore identification of relevant information from the point of view of knowledge base enrichment. As a result, research efforts towards Information Extraction and Natural Language Processing Frameworks have increased significantly over the past decade, Named Entity Extraction and Linking (NEEL) Frameworks being one of the very prevalent ones.

Numerous NEEL frameworks exist today, however, mostly for commercial purposes. The orchestration of components of a NEEL framework, i.e., named entity recognition component, named entity disambiguation and named entity linking component, for microblogging platforms such as Twitter and Facebook is difficult in particular due to the type of text under consideration. As a result, there is little research in the use and improvement of such components towards a more robust framework that can be adapted to emerging information in real time. This thesis discusses the challenges faced by conventional NEEL frameworks when faced with textual formats such as tweets and investigates several approaches to improve the performance of the components and of the NEEL framework as a whole.

A key hypothesis is that the performance of such a framework depreciates

when dealing with social media platforms, and if one component can be used to improve the performance of the other, the overall performance can be improved as well. Supervised and unsupervised techniques have been investigated in this thesis to this end, which prove to be effective in increasing the overall accuracy of the framework when faced with noisy and ambiguous textual formats from the microblogging platform of Twitter.

Italian. L'estrazione di informazioni strutturate a partire dal “web non strutturato”, ha suscitato un notevole interesse da parte delle comunità scientifiche che si occupano di elaborazione del linguaggio naturale e di sistemi basati sulla conoscenza per sviluppare a pieno la visione del “web semantico”. Nell'era moderna, l'uso pervasivo e diffuso delle reti sociali ha portato alla produzione di un flusso continuo di informazioni su piattaforme quali Twitter o Facebook, definite anche piattaforme di microblogging. Tali sorgenti informative, accessibili in tempo reale, producono informazioni caratterizzate dalla presenza costante di rumore e ambiguità linguistiche che rendono particolarmente difficoltoso il compito di estrarre informazioni strutturate. Tale estrazione è tuttavia cruciale per poter arricchire grandi basi di conoscenza, oggi usate in molte applicazioni industriali e di ricerca, con informazioni nuove e rilevanti. Ne risulta che nell'ultimo decennio sono aumentati significativamente gli sforzi della ricerca nel campo dell'elaborazione del linguaggio naturale per l'estrazione di informazioni da piattaforme di microblogging, con particolare attenzione nei confronti dell'estrazione e identificazione di entità nominali (anche Named Entity Extraction and Linking o NEEL).

Oggi giorno esistono numerosi sistemi di NEEL, di cui la maggior parte è creata a scopo commerciale. La calibrazione dei componenti di un sistema di NEEL, cioè dei componenti per la rilevazione, la disambiguazione e l'identificazione di entità nominali, nel caso di piattaforme di microblogging come Twitter e Facebook è difficile in particolare a causa delle tipologie di testo considerato. Mancano approcci di ricerca sistematici volti a guidare l'utilizzo e il miglioramento di tali componenti, per la realizzazione di sistemi più robusti, in grado di meglio adattarsi all'emergere di nuove informazioni, e nuovi interessi (ad esempio, a estrarre tipi di entità nuovi rispetto a quelli considerati in passato). La presente tesi discute le sfide affrontate dai sistemi tradizionali di NEEL qualora questi si misurino con formati di testo quali i tweet, ed esplora vari approcci per migliorare le prestazioni dei singoli componenti e di un sistema di NEEL nel suo insieme.

L'ipotesi chiave del presente lavoro di tesi è che sia possibile costruire sistemi robusti usando dove possibile, componenti esistenti, e che la prestazione di un sistema nel suo complesso possa essere migliorata qualora si sviluppino meccanismi di feedback atti a fare sì che alcuni componenti vengano usati

per migliorare le prestazioni di altri componenti. A tale scopo, in questa tesi sono state indagate tecniche supervisionate e non supervisionate che si sono rivelate efficaci per aumentare l'accuratezza di un sistema nel suo complesso mediante meccanismi di feedback e di adattamento a nuovi domini, per formati di testo ambigui e rumorosi provenienti dalla piattaforma di microblogging Twitter.

TABLE OF CONTENTS

List of Figures	iii
List of Tables	iv
1 INTRODUCTION	1
1.1 Overview	1
1.2 Knowledge Bases at a glance	6
1.3 Organization of the thesis	9
2 PROBLEM DESCRIPTION & LITERATURE REVIEW	10
2.1 Features of Information Extraction	11
2.2 Information Extraction from Twitter	15
2.3 Named Entity Extraction and Linking Frameworks	17
2.3.1 Named Entity Recognition	18
2.3.2 Ontologies for Named Entity Classification	25
2.3.3 Named Entity Linking	29
3 A NEEL FRAMEWORK	38
3.1 Overview of the Proposed Framework	39
3.2 Linking2Adapt	41
3.2.1 Entity Mention Re-Classification	41
3.2.2 Entity Mention Re-Scoping	47
3.3 Learning2Adapt	50
3.3.1 Ontology Mapping	51
3.3.2 Motivation	54
3.3.3 Problem Formulation	57
3.3.4 Contextual Evidence for Learning2Adapt	62
3.4 Learning2Link	64

4	EXPERIMENTAL ANALYSIS	68
4.1	Experimental Analysis: Linking2Adapt	68
4.1.1	Entity Mention Re-Classification	68
4.1.2	Entity Mention Re-Scoping	79
4.2	Experimental Analysis: Learning2Adapt	83
4.2.1	Experimental Settings	83
4.2.2	Experimental Evaluations	88
4.3	Experimental Analysis: Learning2Link	100
4.3.1	Experimental Settings	100
4.3.2	Experimental Evaluation	101
5	CONCLUSION	109
5.1	Future Work	110
6	LIST OF PUBLICATIONS	112
	Appendices	114
A	Experimental Results: Learning2Adapt	114
A.1	Capabilities	114
A.2	Precision, Recall, F-Measure & STMM	115
A.3	Class Wise Accuracy Contribution	116
A.4	Class Wise Precision / Recall	117
B	Learning2Link	119
B.1	Learning2Link for Italian Language Tweets	123
	Acknowledgements	126
	Bibliography	128

List of Figures

1.1	News Article Snippet	2
1.2	Components of a Knowledge Base	7
1.3	LOD Cloud	8
2.1	Information Extraction	12
2.2	A NEEL Framework	18
2.3	Named Entity Recognition: NewsWire Text vs Tweets	19
2.4	Example: Synonymous Entity Mentions	24
2.5	Named Entity Linking: NewsWire Text vs Tweets	30
3.1	NEEL Framework	38
3.2	Proposed Approaches	40
3.3	Linking2Adapt	42
3.4	Manual Mappings between two Ontologies	52
3.5	Example: Ambiguous Entity Mentions	56
3.6	Example of Input-Output Space in Learning2Adapt	58
3.7	Example of Input Space in Learning2Link	65
4.1	Entity Mention Re-classification	69
4.2	Identification Scenarios	71
4.3	Classification Scenarios	74
4.4	Linking Scenarios	76
4.5	Entity Mention Re-Scoping	79
4.6	Example of entity mentions in the Gold Standard	85
4.7	Learning Curves	99

List of Tables

2.1	State-of-the-art NER approaches	22
2.2	State-of-the-art NEL approaches	33
2.3	State-of-the-art NEEL approaches and frameworks	36
3.1	Examples of mappings between T-NER and DBpedia Ontologies	47
4.1	Type-Wise Distribution (%): Ground Truth vs T-NER	69
4.2	T-NER: Identification Performance Analysis	70
4.3	T-NER: Classification Performance Analysis	72
4.4	Entity Linking Performance Analysis	75
4.5	Comparative Analysis: T-NER and T-NER+.	78
4.6	Examples: Re-classification of Entity Mentions	79
4.7	Type-Wise Distribution (%): Training & Dev Ground Truth .	80
4.8	Performance: Entity Identification	81
4.9	Performance: Entity Linking and Classification	82
4.10	NER Oracle: Entity Linking Performance	83
4.11	Dataset Statistics	84
4.12	GS Type Distribution (%) according to Source Ontology (W.C.)	86
4.13	GS Type Distribution (%) according to Source Ontology (Wo.C.)	87
4.14	GS Type Distribution according to Target Ontology (%) . . .	87
4.15	Learning2Adapt Capabilities: Train 2015 & Train 2016	89
4.16	Learning2Adapt Capabilities : Dev, Test 2015 & Dev, Test 2016 (Wo.C.)	92
4.17	Precision, Recall, F-Measure & STMM: Train2015	93
4.18	Precision, Recall, F-Measure & STMM: Train2016	93
4.19	Class Wise Accuracy Contribution (%): Train2015	94
4.20	Class Wise Accuracy Contribution (%): Train2016	94
4.21	Precision, Recall, F-Measure & STMM: Test2015 (W.C.) . . .	94
4.22	Precision, Recall, F-Measure & STMM: Test2016 (W.C.) . . .	95
4.23	Class Wise Accuracy Contribution (%): Test2015 (W.C.) . . .	95

4.24	Class Wise Accuracy Contribution (%): Test2016 (W.C.)	96
4.25	Class Wise Precision / Recall: Train2015 (W.C.)	97
4.26	Class Wise Precision / Recall: Train2016 (W.C.)	97
4.27	Class Wise Precision / Recall: Test2015 (W.C.)	97
4.28	Class Wise Precision / Recall: Test2016 (W.C.)	97
4.29	#Microposts2015: F-Measure (5 instances)	102
4.30	#Microposts2016: F-Measure (5 instances)	102
4.31	#Microposts2015: F-Measure (5 instances) - With Oracle	103
4.32	#Microposts2016: F-Measure (5 instances) - With Oracle	103
4.33	Performance Analysis: Caliano et al. vs L2L	104
4.34	#Microposts2015: SLM (5 instances)	106
4.35	#Microposts2016: SLM (5 instances)	106
4.36	#Microposts2015: SLM (5 instances) - With Oracle	107
4.37	#Microposts2016: SLM (5 instances) - With Oracle	108

1 INTRODUCTION

1.1 Overview

Large volumes of unstructured information is available on the Web today in the form of news articles, research papers, social media posts, business and medical records and so on in human-understandable formats, which are often difficult to manipulate and query by computer applications. The Semantic Web vision has been to express this information in an unambiguous, machine understandable structured format. By doing so, not only is the information organized in a more relevant way for an end-user, but is also easier to manipulate by a computer application for future purposes. Subsequently, Knowledge Base experts today continuously extract information in real-time such as information pertaining to product launches, terrorist strikes, natural disasters, movie celebrities and so on from the Web and organize them in a canonical format for relational databases or knowledge bases that can be queried by an end-user in the future.

Consider the snippet of a news article shown in Figure 1.1 as an example of unstructured text, which consists of several relevant information pieces (as highlighted) spread throughout the article. Since, the article is in a natural language format, it cannot easily be manipulated by a machine unless the relevant information is identified, extracted and organized in a structured format. The process of identifying and extracting relevant information, such as *named entities*, *events* and *relations*, from an unstructured piece of text is known as **Information Extraction** (IE), where:

- a **named entity** signifies an object of relevance in the real world and the process of recognizing text phrases referred to as **entity mentions** (which denote named entities in the real world) is known as **Named Entity Recognition** (NER). For instance, the text phrases *J.K.Rowling*, *Fantastic Beats and Where to Find Them*, *Harry Potter*

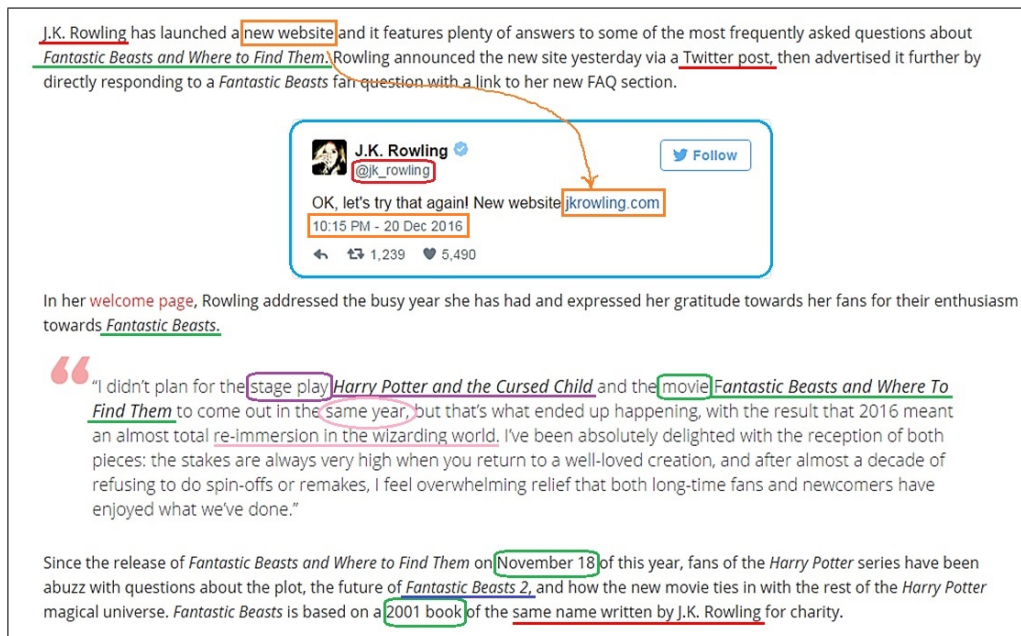


Figure 1.1: News Article Snippet

and the Cursed Child in Figure 1.1 are entity mentions, which denote the real world named entities as their namesake.

- an **event** is descriptive of an action of significance and is usually defined by an actor (or a named entity), an event phrase (action description), a time period and type of action (such as a ceremony), and the process of identifying and extracting events from a piece of text is known as **Event Extraction**. For instance, the text phrase ‘jkrowling.com’ seen in Figure 1.1 denotes an event that can be defined by means of $\{jkrowling.com, launched, 20 Dec 2016, Product Launch\}$.
- a **relation** is defined by an association among entity mentions in a piece of text and the process of identifying the association between, say, two entity mentions is known as **Relation Extraction**. For instance, the relations $genre(\text{Harry Potter and the Cursed Child}, \text{Stage Play})$; $releaseDate(\text{Fantastic Beasts and Where to Find Them}, 18 \text{ November } 2016)$ can be observed and extracted from Figure 1.1.

In the modern digital age, social media platforms such as Facebook¹, Twitter², LinkedIn³, and Instagram⁴ have become a user-friendly and an easily accessible means for an end-user to communicate with others (such as share announcements, and opinions) in real-time. For instance, in Figure 1.1, the author *J.K.Rowling* reveals the name of her new website by using Twitter. The elaborate and extensive use of these platforms has led to constant availability of *user-generated content*. As a result, such microblogging platforms have become a source of constant attention of researchers to constantly extract new information from unstructured text readily available from these platforms. Twitter, in particular, has witnessed an increasing interest in the scientific community given its 140-character format, commonly referred to as tweets, which restricts users to convey information or express an opinion in a concise, yet meaningful way. Consequently, use of Internet slangs, acronyms, hashtags, shortened URLs, emoticons and GIFs has become quite a commonplace today. Essential information such as named entities, opinions and events are abbreviated, #hashtagged or denoted by @*usernames* so as to be able to utilize the character space efficiently. For instance, a new entity mention (one which has not yet been extracted and indexed in a knowledge base) `jkrowling.com` is observed in the tweet in Figure 1.1. While knowledge bases such as DBpedia⁵ already contain information about the author *J.K.Rowling*, the new entity mention as well as a new relation with regards to the new mention `website(J.K.Rowling, jkrowling.com)` can be extracted and added to the author’s DBpedia page⁶.

Thus, the task of Named Entity Recognition (NER), which comprises of identifying entity mentions (also known as **Named Entity Identification** or NEI) from an unstructured piece of text, as exemplified above and subsequently classifying them according to a given domain *ontology* (also known as **Named Entity Classification** or NEC), which is usually defined as a set of disjoint types [51], is essential to the other tasks of Information Extraction

¹<https://www.facebook.com/>

²<https://twitter.com/>

³<https://www.linkedin.com/>

⁴<https://www.instagram.com>

⁵<http://wiki.dbpedia.org/>

⁶http://dbpedia.org/resource/J._K._Rowling

(viz, Event Extraction [110,128] and Relation Extraction [59]), as well as for applications, which include political campaign analysis [124,126], sentiment analysis [66,91], and product/news recommendation [94], to name a few. According to [49,52], an **ontology** can be described as a conceptual system underlying a particular Knowledge Base (KB), based on which KBs can be differentiated by use of different ontologies. In simplistic terms, it refers to a classification hierarchy or a taxonomy for a particular domain with classification types according to the domain in consideration. For instance, an ontology related to a *music domain* will consist of classes or classification types such as *musicArtist*, *releaseDate*, *label*, *recording*, *instrument* and so on. Further the task of **Named Entity Linking** (NEL) deals with grounding an identified entity mention to a resource in a Knowledge Base such as Wikipedia, DBpedia, YAGO, or Freebase, wherein a KB describes real-world objects (i.e., the named entities), referred to by corresponding entity mentions, by specifying their types and their relations with other real world objects. Collectively, the tasks of Named Entity Recognition and Named Entity Linking are being studied under the **NEEL** Problem i.e., **Named Entity Extraction and Linking**, for different domains, languages and textual formats. A survey of state-of-the-art systems or frameworks dealing with the NEEL problem is provided in Chapter 2.

Essentially, a NEEL framework is composed of an entity recognition component (responsible for the processes of entity identification and classification), a disambiguation and a linking component. A detailed description of these components is provided in Chapter 2, along with an exhaustive study of state-of-the-art approaches behind these components. In order for a NEEL framework to achieve optimal performance, one might argue the simplest solution would be an assembly of the best performing components in the state-of-the-art as a framework. However, there are several constraints to such a solution. For instance, a recognition component, which performs optimally for long textual formats may not be suitable for short microblogging textual formats or a linking component, which is able to ground entity mentions pertaining to a music domain, may not be able to link mentions belonging to a medicine domain. Thus, the main goal of this thesis is the study of the components of a given NEEL framework by investigating various scenarios in

such a way that one component can be used to improve the performance of the other, thus optimizing the overall performance of the framework. These scenarios have been investigated under three major case studies (defined in Chapter 3) while addressing the following research questions:

- **RQ1:** Performances of NEEL frameworks suffer when it comes to entity identification and linking in microblogging environments. A plausible research question here is what all forms of evidence can be used in order to strengthen the performance of the framework? and,
- **RQ2:** Given the existence of multiple domain-centric entity recognition systems today, is it possible to adapt an entity recognition system based on different ontologies instead of re-training a system every time a new ontology is introduced?

The afore-mentioned research questions have been studied in this thesis and addressed by the contributions as summarized below:

1. **LinkingToAdapt:** use of the linking component of a NEEL framework for:
 - an unsupervised re-classification approach to adapt named entity classification performed by a recognition component based on the linking component;
 - an unsupervised re-scoping approach to adapt named entity identification performed by a recognition component based on the linking component.
2. **LearningToAdapt:** use of a supervised re-classification approach to adapt named entity classification performed by a recognition component in a NEEL framework for different ontologies (used by different named entity classifiers) inspired by a transfer learning paradigm based on the entity mentions identified by the recognition component.
3. **LearningToLink:** use of a supervised entity linking approach to adapt the linking component used for linking entity mentions identified from a given text based on the recognition and disambiguation components in a NEEL framework.

Contributions 1 and 3 have been provided to address RQ1, while Contribution 2 addresses RQ2. For conducting the afore-mentioned case studies, a NEEL framework has also been proposed for the microblogging platform of Twitter, which is also discussed in detail in Chapter 3. Experimental analysis of the framework using different tweet datasets for the contributions is further provided in Chapter 4.

1.2 Knowledge Bases at a glance

Resources such as knowledge bases, vocabularies, dictionaries and encyclopedias have been used for the better part of this century as references to the true facts or meanings of millions of real world objects. While encyclopedias provide summaries of real world objects by means of unstructured, long textual (though human-comprehensible) formats, knowledge bases have evolved to be more structured so as to be more machine-comprehensible. As stated in [6], a KB comprises of two main components: the TBox and the ABox. The TBox is used for introducing the *Terminology*, i.e., the *concepts* in a KB and *roles* denoting binary relationships between these concepts, as seen in Figure 1.2(a), where *Organization*, *Thing*, *Agent*, *Person*, and *Park* are concepts and, in particular, the concept *Organization* is defined by the role *hasEmployee*. Further, ABox is used for defining *assertions* of particular named individuals which are, in fact, *instances* of a given concept. For instance, in Figure 1.2(b), the instance *IBM* is an assertion of the concept *Organization* and is further defined by assertions such as *hasName*, *foundedBy* and so on. In other words, an ABox is used for specifically defining instances of concepts using specific set of roles.

One of the biggest and most widely used multi-lingual encyclopedia is **Wikipedia**⁷ consisting of information about millions of real world objects in over 250 distinct languages. While Wikipedia mostly provides content in an unstructured format, there are certain levels of structuring to it as well, i.e., the use of infoboxes, tables and lists. However, in spite of the presence of such structuring, interpretability and inferencing from Wikipedia has always been

⁷<https://en.wikipedia.org/wiki/>

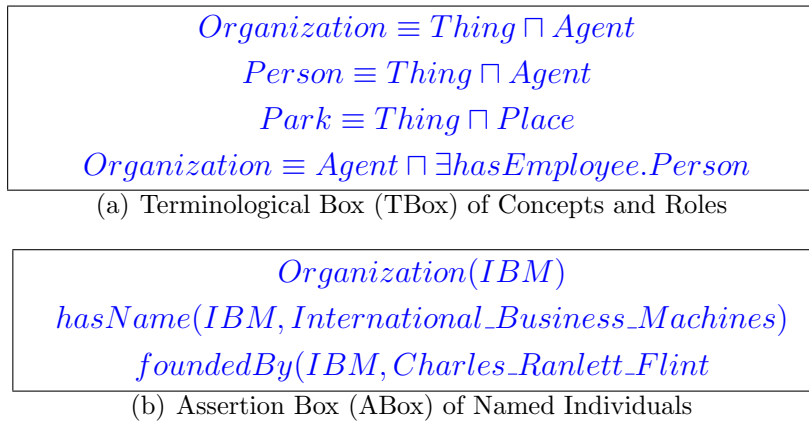


Figure 1.2: Components of a Knowledge Base

difficult. For instance, one can easily find a Wikipedia list of all presidents of the United States and another Wikipedia list of birth dates of all the presidents of United States, however, it is difficult to find a Wikipedia list of all presidents who are older than, say, *George Bush*. As a result, several structured knowledge bases have come into existence, which use Wikipedia as a source to extract information, **DBpedia** being one of the most popular and generic (multi-domain) one among them.

DBpedia uses an information extraction framework for information available from Wikipedia infoboxes [4], and currently describes over 4.5 million things in ≈ 125 languages⁸. Every named entity is represented in DBpedia by its corresponding article name from Wikipedia. Further, every entity is associated with a set of properties such as religion, nationality, spouse and so on and classified using four different classification schemas, i.e., DBpedia Ontology, Wikipedia Categories, YAGO and UMBEL. The DBpedia Ontology⁹ is known to be a complex, detailed and a cross-domain ontology with 685 classes, described by 2,795 properties¹⁰. Classes in DBpedia are described by properties. For instance, the class *Person*¹¹ is described by properties such as *birthName*, *age*, *bloodGroup* etc. DBpedia has been used as a referent KB

⁸<http://wiki.dbpedia.org/about/facts-figures>

⁹<http://mappings.dbpedia.org/server/ontology/classes/>

¹⁰<http://wiki.dbpedia.org/services-resources/ontology>

¹¹<http://mappings.dbpedia.org/server/ontology/classes/Person>

for the experimental analysis in this thesis.

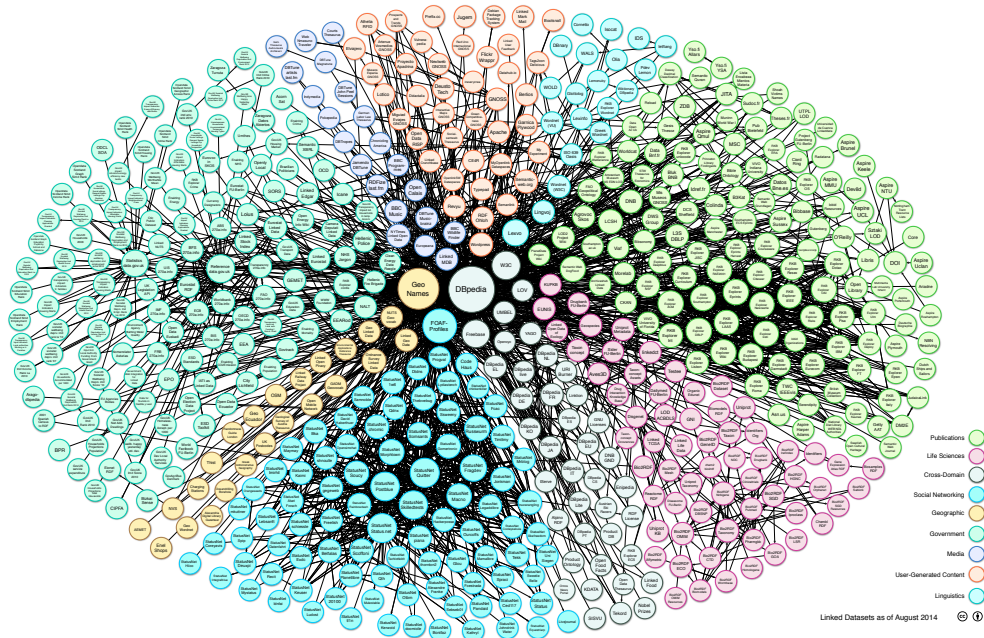


Figure 1.3: LOD Cloud

Over the last two decades, with a rapidly evolving technology, scientists have quickened the pace of generating machine-understandable information structures, i.e. by constructing knowledge bases, from information which was originally used for human consumption as an effort to realize the goals associated with the Semantic Web RoadMap¹², the term ‘Semantic Web’ being coined by Sir Tim Berners-Lee. Furthermore, the **Linked Data Community** has come to be as a result of publishing and connecting structured data available on the Web using RDF (Resource Description Framework) formats used for representing entities and their URIs on the web [13], thus giving way to the Web of Data. By doing this, different KBs/datasets (with different ontologies) can be linked on the Web, moving from generic ontology types to domain-specific ones. These datasets are visualized as the *Linked Data Cloud* or *LOD Cloud*, a depiction of the same is shown in Figure 1.3¹³ (as per August 2014) where each node in the cloud represents a distinct dataset

¹²<https://www.w3.org/DesignIssues/Semantic.html>

¹³<http://lod-cloud.net/>

published as Linked Data, and arcs represent RDF links between two linked datasets. While the figure clearly depicts a huge number of linked datasets available today and is a positive step towards the vision of the Semantic Web, an underlying observation is that these datasets are characterized by varying, independent and more often than not, domain-specific or language-specific ontologies. Significant approaches in the state-of-the-art have been proposed towards domain adaptation or ontology mapping [34, 35], wherein, classification types of one ontology are adapted to classification types of another ontology. While such approaches are useful for cross-domain applications, the text under consideration is mostly overlooked in such cases. For instance, textual formats such as microblogs, where entity mentions can be classified into types ranging from generic ones such as (*Person*, *Product*, or *Location*) to specific ones such as (*Population*, *Latitude*, or *Longitude*). In such cases, an ontology mapping criterion of simply mapping the types of one ontology to the types of another (i.e., concept-based mapping) may not suffice. This is because of the usage of natural language, the concise nature as well as the presence of ambiguity in such texts. For such cases, a unique instance-based mapping (as opposed to the conventional concept-based mapping) between the types of a source and target ontology has been studied in this thesis, which takes into consideration not only the classification type that has to be mapped from a source ontology to the target ontology, but also the entity mention and its contextual usage under consideration.

1.3 Organization of the thesis

An introduction to the problem and the main contribution of the thesis have been summarized in this chapter. Chapter 2 provides a detailed account of the problem introduced in Chapter 1 along with an exhaustive survey of the state-of-the-art approaches. Further, a NEEL framework which has been proposed for microblogging environments and the use of this framework as a means to establish the contributions of this thesis have been presented in Chapter 3. The experimental results that support the proposed approaches have been provided in Chapter 4. Finally, concluding remarks and future directions are outlined in Chapter 5.

2 PROBLEM DESCRIPTION & LITERATURE REVIEW

The thirst for knowledge discovery from the unprecedented amounts of data easily available today (in the form of online news, social media messages and posts, scientific documents and so on) drives researchers to perform the tasks of Natural Language Processing (NLP) and extract relevant information, wherein NLP deals with linguistic analysis of user-generated content (text or speech) in any natural language and Information Extraction techniques allow researchers to seek meaningful patterns for knowledge discovery by extracting relevant structured text from an unstructured text. This chapter presents a detailed discussion on Natural Language Processing and Information Extraction techniques, as well as an account of the state-of-the-art approaches proposed for the same. Further, the NEEL problem and components of a NEEL framework are described in detail, along with the state-of-the-art approaches proposed for these components.

IE tasks such as Named Entity Recognition, Relation Extraction, Event Extraction and so on have been used over the years to eventually extract information in the form of named entities, relations between named entities, and events so as to enable tasks such as text summarization, semantic Web searches, personalized recommendations, and enrichment of knowledge bases. Interest in the research community towards NLP and Information Extraction started to peek with the regular occurrence of the Message Understanding Conferences (MUC) year after year [51]. These conferences, organized by DARPA, were initiated with the goal of Information Extraction primarily in the late 1980s. In fact, the task of Named Entity Extraction (mostly also known as Named Entity Recognition) was introduced in the Sixth Message Understanding Conference (MUC-6) where the term ‘Named Entity’ was coined for the first time [51]. Henceforth, research in this field has come a long way from the use of handcrafted rule-based algorithms [3, 100] to the use of sophisticated machine learning techniques [112, 122].

While essentially the crux of an information extraction technique is to be able to process a piece of text and extract relevant information in a structured format, the underlying text being processed highly influences the technique being used. For instance, if the given text is already structured with, say, tabular information, then a mere set of handcrafted rules might suffice the process of extracting relevant information. While on the other hand, if the given text is in an unstructured free-style format, then several steps of pre-processing (such as syntactic and semantic analysis) might be required in addition to the simplistic rules at hand. Other features such as the language of the text (Arabic vs English), grammatical constructs, and domain of the text (Medicine vs Science Journals) also play a key role in determining the techniques being employed. A brief description about these features, and how they influence the methods and techniques being used for extracting information is provided in the next section.

2.1 Features of Information Extraction


Several features such as the language of a text, the genre of the text and domain of the text need to be taken into consideration while using an information extraction approach to extract entity mentions, events or relations. These features as well as state-of-the-art approaches proposed specifically keeping in mind these factors are summarized as below:

1. ***Language Specific Information Extraction Approaches:*** Different languages practice different grammatical and syntactical rules as well have different writing styles. As a result, while an approach for an IE task may be efficient for extracting information (entity mentions or events or relations) from an English language text, it might not perform well for texts in, say, Spanish or Greek. A variety of multilingual [89,90] as well as language-specific IE systems exist today for extracting relevant information from languages such as Greek, Spanish, Hindi, French, Arabic, Italian and so on [12,72,85]. In this thesis, English language texts have been used for experimental analysis for the proposed approaches, however, an additional language (Italian) is also

experimented with.

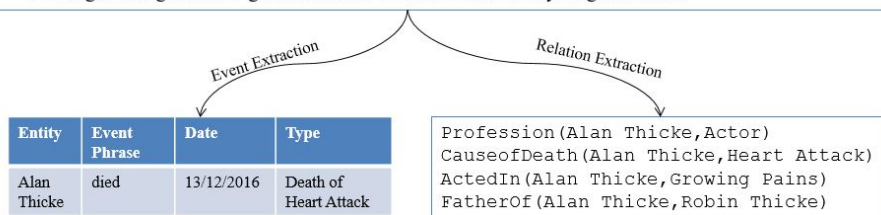
2. **Genre Specific Information Extraction Approaches:** Today IE techniques are used for extracting information from a variety of different genres of unstructured textual formats, varying from news articles [115], medical records [58,120], scientific journals, emails [76,82], private user messages (SMS) [36] and government documents to social media posts [71,74,102,103]. The genre of a text essentially also determines the textual format being used. For instance, a scientific journal or a news article will use long, formal and descriptive textual formats in natural languages, while social media posts will mostly use short and informal textual formats, with colloquial expressions.

Actor [Alan Thicke](#) has died at the age of 69.



[Thicke](#), known for his role as the likable father on the [ABC](#) television series [Growing Pains](#), died from a heart attack on Tuesday.

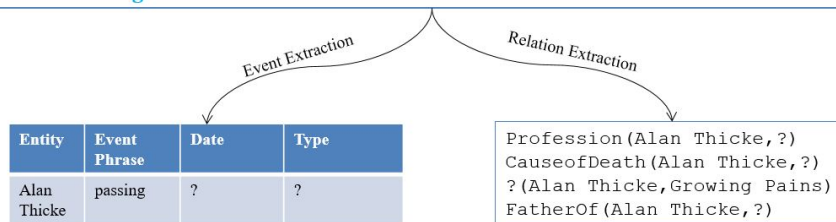
His son, [Grammy award](#) nominated-singer [Robin Thicke](#) paid tribute to him on Tuesday evening, calling him the 'greatest man I ever met' and 'always a gentleman.'



(a) Information Extraction from (Long) Unstructured Text



So saddened to hear of the passing of my friend [#AlanThicke](#) Sending my 🙏 2 his loved ones! ❤️
[#RIP](#) [#GrowingPains](#)



(b) Information Extraction from (Short) Unstructured Text

Figure 2.1: Information Extraction

Two different unstructured textual formats are shown in Figure 2.1 where Figure 2.1(a) represents (long) text from a news article and Figure 2.1(b) represents (short) text from a social media post, i.e., a tweet. Both texts have been posted by end-users for communicating about an important news, i.e., the death of actor ‘Alan Thicke’. The entity mentions have been highlighted in blue, while the events and relations that are extracted from these texts are also shown in the figures. An important observation here is that when it comes to tasks such as entity recognition, event extraction and relation extraction, short text pieces can be very challenging due to absence of enough context to decipher the relevance of information being extracted, in contrast to long texts. For instance, the task of entity recognition from the tweet in Figure 2.1(b) can be difficult if text pre-processing (such as removal of #hashtags) is not done. Further, as opposed to Figure 2.1(a), event and relation extraction also seem difficult in Figure 2.1(b) due to not enough information available in the post, as well as lack of context. While, on one hand, microblogging platforms have become highly popular today in the research community since they constantly experience the emergence of new information, on the other hand, they present many challenges in the form of ambiguity, noise and lack of context. A description of the state-of-the-art approaches which deal with short textual formats is provided in Section 2.2 along with an in-depth discussion of the challenges associated while dealing with short texts obtained from microblogging platforms.

3. ***Domain Specific Information Extraction:*** Knowledge about a domain (such as medicine, music, government documents and so on) from which information has to be extracted usually helps in evaluating the pre-requisites for an IE task. For instance, if entity mentions are being extracted from a medicine domain, a medicine oriented ontology with classification types such as PROTEIN, CELL-TYPE, DNA [111] will be preferred over a generic ontology with classification types such as PERSON, LOCATION, ORGANIZATION, PRODUCT. Similarly, information in terms of events [129] and relations [17] specific to the

medicine domain will be searched and extracted. Additionally, the prevalence of domain-specific knowledge bases (such as MusicBrainz¹, which is an open music encyclopedia, GeoNames², which is a geographical database) in modern day and age leads to the encouraged use of domain-specific information extraction approaches, which further help in simulating other NLP tasks such as named entity linking. Two different ontologies have been studied in this thesis [102, 107] (in Chapter 3), which are used for classification of entity mentions discovered from short textual formats.

Considering the features stated above, several methods for performing an information extraction task have been proposed in the state-of-the-art ranging from rule-based methods (use of handcrafted rules by human experts to extract information) [3, 100] and pattern-based approaches (use of patterns for information extraction) [83] to machine learning (unsupervised, semi-supervised and supervised) ones. Over the years, different learning approaches have become prevalent for information extraction from unstructured text, as discussed briefly below:

- *Unsupervised machine learning:* Let $X = (x_1, x_2, \dots, x_n)$ be a random vector with n independent observations. In an unsupervised learning task, the goal is to look for a structure or pattern in X , which is accomplished by techniques such as clustering data into multiple groups/clusters, and detecting outliers. Unsupervised learning approaches have been used in the state-of-the-art for entity recognition tasks [22, 37, 86] from different textual formats.
- *Supervised machine learning:* Let $X = (x_1, x_2, \dots, x_n)$ be a set of input instances (or feature vectors) and $Y = (y_1, y_2, \dots, y_m)$ be a set of classification labels such that $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_m)$ constitute a training data set. The goal of a supervised machine learning approach is to be able to predict the classification label of an unlabeled instance based on what the model has learnt from the training data

¹<https://musicbrainz.org/>

²<http://www.geonames.org/>

set. Numerous supervised learning approaches such as Conditional Random Fields (CRF, which is an undirected graphical model) [69], Hidden Markov Models (HMM, which is a generative model) [48], Support Vector Machines (SVM) [24], Maximum Entropy Markov Model (MEMM, which is a discriminative model) [78] and so on have been used in the state-of-the-art for the IE tasks of NER [79, 102, 134] and relation extraction [133].

- *Semi-supervised machine learning*: This approach lies halfway between supervised and unsupervised learning, in the sense that from a given set of instances X , some part of the instances X_a (also referred to as *seed instances*) have their corresponding labels, while for the other part of the instances X_b , the labels are not known [135]. The goal of the system essentially is to predict the labels of instances in X_b based on the learnt model from X_a and iteratively use this knowledge to predict the labels of new instances. One of the well-known semi-supervised learning approach is *bootstrapping* [67, 101] whereas other features such as use of word representations [125] have also been explored in the state-of-the-art to enable entity recognition in a semi-supervised manner.

Both unsupervised and supervised learning approaches have been used in this thesis to conduct the case studies for short texts retrieved from Twitter in English and Italian languages. A description of state-of-the-art approaches for information extraction from the Web as well as from microblogging platforms is presented in the next section.

2.2 Information Extraction from Twitter

Social media platforms have provided a means to the user community to express opinions and sentiments, announce important events and just talk about anything virtually. As a result, over the years these platforms have witnessed a steady growth in their user communities. For instance, Facebook today has a community of 1.79 billion monthly active users (as of September 30, 2016) from 1 million total active users (in December 2004)³. On the other

³<http://newsroom.fb.com/company-info/>

hand, Twitter has a constantly growing community of 313 million monthly active users (as of June 30, 2016) where over 500 million tweets are being sent daily (as of August 2013)⁴. While both platforms are equally popular among people and used extensively, Twitter has attracted the attention of the research community due to its ease of access, which allows end users to use natural language expressions and special characters such as @, # and colloquial constructs. As a result, usage of such expressions has become quite popular in other microblogging platforms (such as Facebook, Instagram) as well. Another reason for the popularity of Twitter is the 140-character limit⁵ due to which many users try to make the best of the available space (by using abbreviations, and keeping noise at a minimum with strong subject matter), while others have seen to be managing the character space by the use of unnecessary punctuation marks, GIFs and emoticons without actually tweeting something substantial.

In spite of Twitter's popularity among end-users, which makes it highly relevant for the research community, there are certain challenges associated as well while dealing with such textual formats, as summarized as below:

- **Conciseness of text:** The limited character space, which makes it a preferable means of communication among the end users, is also one of the biggest concern when dealing with microposts obtained from Twitter. This is so because, although concise, tweets can be rich of embedded semantics. While addressing entity recognition, event extraction or relation extraction from such a platform, it is necessary to bridge the semantic gap between the few words written by a user and the corresponding, more complex, meaning.
- **Noisy content:** Tweets are characterized by the use of colloquial expressions, abbreviations, emoticons, word shortening, irregular capitalization, and emphatic expressions. Due to limitations on the blog-post size, their compliance to canonical grammatical rules suffers as well. An additional aspect that should be explicitly modelled while dealing

⁴<https://about.twitter.com/company>

⁵Lately, a workaround has come up for this restriction for re-tweets, where a user can add his/her own comments to the re-tweeted text, which can easily extend 140-characters.

with such microblog posts, thus, relates to bad-formed texts, where vocabulary, spelling and syntax represent a linguistic challenge.

- **Dynamics:** Microblog contents are characterized by a strong temporal dynamic due to the continuous evolution of trending topics as well as by their potential to open a debate with contents provided by other users. Thus, the focus on specific types of entity mentions of interest considered by a NER system may evolve over time.

Due to the afore-mentioned challenges, information extraction from the microblogging platform is still relatively new and difficult, as opposed to information extraction approaches from long textual formats. Although various approaches have been proposed in the state-of-the-art for tackling the tasks of entity recognition [71, 74, 102] and event extraction [103, 128] from Twitter, there is still room for improvement, given the idiosyncratic nature of tweets and the constant emergence of new information on such a platform.

2.3 Named Entity Extraction and Linking Frameworks

In recent times, Named Entity Extraction and Linking (NEEL) frameworks have become popular in the Computer Science community for performing the tasks of entity recognition, followed by entity linking. A conventional NEEL framework, as shown in Figure 2.2, is essentially composed of a recognition component (and performs NEI and NEC), a disambiguation component and a linking component. A basic form of text pre-processing (before proceeding with the recognition component) is not uncommon these days so as to remove noise (in the form of special characters or emoticons) and perform text segmentation. As the name suggests, in segmentation, a text is segmented into a set of text chunks, wherein a segment may be a meaningful string of words that occur mostly together, or is an entity mention. A segment attributing to an entity mention is defined by IOB encoding, which means that a word is the beginning (B), inside (I) or outside (O) of the entity mention [97]. For instance, consider the tweet and its segmented version as shown below:

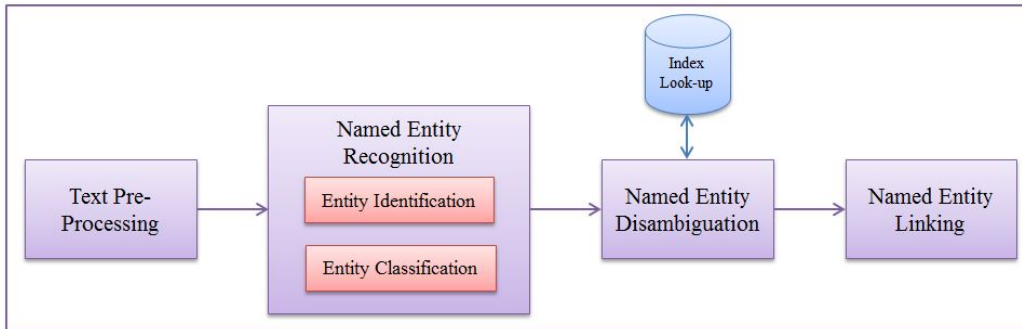


Figure 2.2: A NEEL Framework

Just been to the Top of Empire State building. That's awesome man!

⇓ segmented as

(Just_o been_o to_o) | (the_o Top_o of_o) | (Empire_{B-Entity} State_{I-Entity} building_{I-Entity}) | . (That's_o awesome_o) | (man_o)!

Other forms of text pre-processing includes POS (part of speech) tagging wherein a part of speech annotation (such as noun, verb, adjective) is assigned to every word in the text under consideration. These pre-processing tasks help a recognition component in identifying an entity mention and its type with a higher precision. The components of a NEEL framework and approaches proposed for these components in the state-of-the-art have been described in detail in this section. Finally, popular NEEL frameworks that have been proposed for microblogging platforms have been discussed in the end.

2.3.1 Named Entity Recognition

Over the years, the task of recognizing entities has been extensively addressed by the research community in varying scientific fields ranging from medicine to economics; from different textual formats and languages ranging from long unstructured textual records to today's ill-formed short texts and using a variety of platforms, systems and tools as described in detail in the previous

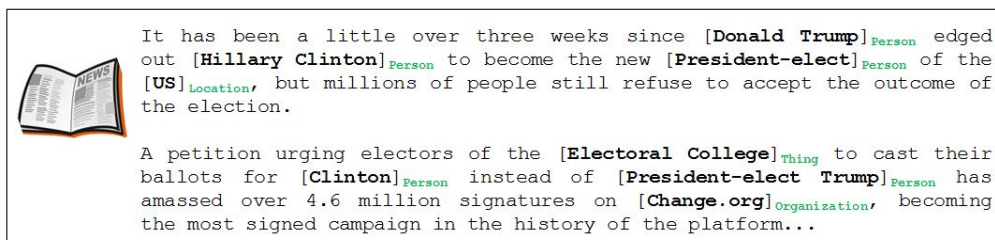
chapter. In general, the task of Named Entity Recognition can be categorized by two important subtasks, as follows:

- *Named Entity Identification*, wherein pieces of text pertaining to real world objects are identified as entity mentions;
- *Named Entity Classification*, which deals with the semantics of the identified entity mentions, i.e., the mentions are classified into pre-defined categories (or entity types or classes), such as PERSON, LOCATION, and ORGANIZATION according to an underlying ontology, as defined in Chapter 1.

Please note that the terms *named entity* and *entity mention* have been distinguished (as described in Chapter 1) so as to avoid any confusion between a text phrase recognized as an entity mention and its corresponding real world representation. Figure 2.3 depicts the processes of Named Entity Identification and Classification, as described above; from two different textual formats i.e., a short, ill structured text and a long, semi-structured text; where entity mentions have been identified (highlighted in bold) and classified using a simplistic ontology, which comprises of the entity types: PERSON, LOCATION, ORGANIZATION and THING.



(a) Example: Ill Structured Short Text - Tweet



(b) Example: Semi-Structured Long Text - News Snippet

Figure 2.3: Named Entity Recognition: NewsWire Text vs Tweets

Numerous NER systems have been proposed in the past for long and short textual formats. Generic systems such as ANNIE [26] and Stanford NER [43] have been developed originally for entity recognition from long textual formats such as newswire text and blogs, however, they have also been used for short textual formats, though, with a loss in performance [31] due to the challenging nature of such texts as discussed in the previous section.

In [74], a K-Nearest Neighbour (KNN) classifier is proposed for identifying entity mentions from tweets, which combines a Conditional Random Field (CRF) model using a boot-strapping scheme for semi-supervised learning. The entity mentions are classified into the entity types PERSON, PRODUCT, LOCATION or ORGANIZATION. Further, an NLP based supervised framework has been proposed in [102] for entity recognition from tweets, named T-NER, where tasks such as tokenization, POS tagging and shallow parsing are performed, followed by which topic models (i.e., LabeledLDA) are applied by using conventional features such as orthographic, contextual and dictionary features as well as tweet-specific features such as using retweets, @usernames, #hashtags, and URLs to identify and classify entity mentions from tweets into 10 distinct entity types of BAND, COMPANY, FACILITY, GEO-LOCATION, MOVIE, OTHER, PERSON, PRODUCT, SPORTSTEAM and TVSHOW. In [71], an unsupervised NER system has been proposed for entity recognition from targeted Twitter streams, which takes into account the local context in the Twitter stream by deploying a random walk model, and a global context obtained from Wikipedia and the Web N-Gram corpus. This system does not, however, rely on any linguistic features in order to be independent of specific Twitter streams.

Further, in [21], Cherry et al. (2015) propose a semi-Markov named entity recognizer using word vectors and Brown clusters built in an unsupervised setting in order to deal with unlabelled tweets, to identify entity mentions and classify them as PERSON, LOCATION or ORGANIZATION, by making use of an importance data-weighting scheme that leverages annotated newswire data. A Twitter named entity recognizer has been proposed in [30], which uses unsupervised clustering for feature generation, Freebase gazetteers [5], and ANNIE first name lists [26] while also focusing on the drift (i.e., change in entity mentions over time) introduced in input data.

The mentions are classified into entity types of COMPANY, FACILITY, GEO-LOCATION, MOVIE, MUSICARTIST, OTHER, PERSON, PRODUCT, SPORTSTEAM and TVSHOW. On the other hand, commercial tools such as DBpedia Spotlight⁶ and TextRazor⁷ are available as well, for entity recognition from Twitter. However, [31] evaluate and report poor performances of these tools over state-of-the-art Twitter gold standards as compared to their performance over newswire text. Other details for these and several other approaches w.r.t. the taxonomies and the datasets used for experimental analysis is presented in Table 2.1.

Apart from the challenges mentioned previously when dealing with short texts, ambiguity in an entity mention’s usage is another significant challenge that leads to difficulties in entity classification (discussed in Section 2.3.2) and linking (discussed in Section 2.3.3). Ambiguity in a mention’s usage has been observed in the following forms:

1. **Polysemy:** According to WordNet [39,81], polysemy is defined when a *word form* has more than one *word meaning*. In context of named entities and entity mentions, polysemy means that a text phrase (referring to an entity mention) can denote more than one named entity in the real world. This further signifies that polysemous text phrases may lead to different entity types and should be linked to different KB resources. For instance, consider the following tweets:

- ◇ The first [#**HarryPotter**]⁸_{Film} film turns 15 today. Here’s a look at scandals surrounding the series <http://rol.st/2fPBJJh>
- ◇ Daniel Radcliffe was in the bath when he found out he’d been cast as [**Harry Potter**]⁹_{Character}. He cried.
- ◇ he hasn’t read [**Harry Potter**]¹⁰_{Book} but he thinks he understands politics

⁶<http://dbpedia.org/spotlight>

⁷<http://www.textrazor.com>

⁸[http://dbpedia.org/page/Harry_Potter_\(film_series\)](http://dbpedia.org/page/Harry_Potter_(film_series))

⁹[http://dbpedia.org/page/Harry_Potter_\(character\)](http://dbpedia.org/page/Harry_Potter_(character))

¹⁰http://dbpedia.org/page/Harry_Potter

Table 2.1: State-of-the-art NER approaches

S.No.	Name of the Approach	Learning Methods	Features	Datasets	Domain	Taxonomy
1	Liu et al, 2011 [74]	Semi-supervised learning	KNN classifier with a linear CRF model <ul style="list-style-type: none"> • POS tagging using CRF • Capitalization classifier T-CAP to predict capitalization using SVM • T-SEG for segmentation using IOB encoding by CRF • Classification using distant supervision (LabeledLDA) • Global context using Wikipedia & Web N-Gram corpus for tweet segmentation • Local context- random walk model for each segment • Tokenization-adaptation of ANNIE’s English tokeniser • Normalization-combination of a generic spelling correction dict. & a social media specific dict. • PoS tagging-adapted using Stanford tagger trained on tweets tagged with PTB 	Manually annotated datasets	Twitter	CoNLL, ACE
2	T-NER Ritter et al, 2011 [102]	CRF for entity segmentation & LabeledLDA for entity classification	<ul style="list-style-type: none"> • POS tagging using CRF • Capitalization classifier T-CAP to predict capitalization using SVM • T-SEG for segmentation using IOB encoding by CRF • Classification using distant supervision (LabeledLDA) • Global context using Wikipedia & Web N-Gram corpus for tweet segmentation • Local context- random walk model for each segment • Tokenization-adaptation of ANNIE’s English tokeniser • Normalization-combination of a generic spelling correction dict. & a social media specific dict. • PoS tagging-adapted using Stanford tagger trained on tweets tagged with PTB 	Ritter et al, 2011 [102]	Twitter	Freebase
3	Li et al, 2012 [71]	Unsupervised approach	<ul style="list-style-type: none"> • Global context using Wikipedia & Web N-Gram corpus for tweet segmentation • Local context- random walk model for each segment • Tokenization-adaptation of ANNIE’s English tokeniser • Normalization-combination of a generic spelling correction dict. & a social media specific dict. • PoS tagging-adapted using Stanford tagger trained on tweets tagged with PTB 	Targeted Twitter stream	Twitter	Wikipedia
4	TwitIE Bonicheva et al, 2013 [14]	Open source NLP pipeline	<ul style="list-style-type: none"> • Normalization-combination of a generic spelling correction dict. & a social media specific dict. • PoS tagging-adapted using Stanford tagger trained on tweets tagged with PTB 	Ritter et al, 2011 [102]	Twitter	CoNLL
5	Derczynski et al, 2015 [30]	<ul style="list-style-type: none"> • Structured Learning using CRF, L-BFGS updates • Unsupervised clustering features 	Uni/bigrams to model context, Brown clusters (m>2000), PoS tagging, Freebase gazetteer	<ul style="list-style-type: none"> • W-NUT 2015 [7] • Ritter et al, 2011 [102] 	Twitter	Freebase
6	Karatayet et al, 2015 [65]	Unsupervised approach	Tweet segmentation for NER & user profile model generation (using entity mentions with their frequency counts in a tweet obtained from followers’ posts)	<ul style="list-style-type: none"> • General tweets (100) • Personal tweets (100) [Turkish tweets] • Ritter et al 2011 [102] • Finin et al 2010 [42] • #Microposts2013 [19] • Wordsmith Corpus (SEM’14) 	Twitter	Wikipedia
7	Ohwaseyji et al, 2015 [41]	CrowdSourcing (CrowdFlower-767 workers)	Annotation instructions, disambiguation instructions, entity types information, interface for annotation	<ul style="list-style-type: none"> • Ritter et al 2011 [102] • Finin et al 2010 [42] • #Microposts2013 [19] • Wordsmith Corpus (SEM’14) 	Twitter	(adapted) MUC
8	Cherry et al, 2015 [21]	Brown clusters & word vectors to leverage unannotated data & an importance weighting scheme for annotated data	<ul style="list-style-type: none"> • Semi-Markov Tagger for NER with Passive-Aggressive algorithm as the learning algorithm. • Use of unsupervised word representations (Brown clusters & word vectors) to leverage unannotated tweets • Use of data weighing scheme to leverage annotated newswire data 	<ul style="list-style-type: none"> • Finin et al, 2010, [42] • Ritter et al, 2011 [102] • Fromreide et al., 2014 [46] • Manually annotated corpus <p>Newswire datasets: CoNLL 2003 [123] newswire, training set as a source of out-of-domain NER annotations</p>	Twitter	(adapted) MUC

Here, the entity mention ‘Harry Potter’ (highlighted in bold) in itself is not polysemous, since it denotes a unique named entity in the real world. However, the text phrase is polysemous in terms of its usage and thus denotes different real world named entities, consequently, leading to different entity types and different KB resources, as indicated.

2. **Synonymy**: According to WordNet [39,81], synonymy is defined when a *word meaning* has more than one *word forms*. In context of named entities and entity mentions, synonymy means that one named entity of the real world has been denoted by different text phrases (referring to entity mentions) on social media platforms. In contrast to polysemous mentions, this signifies that synonymous text phrases will have the same entity type and, therefore, should be linked to a unique KB resource. For instance, consider the tweets in Figure 2.4, where one can observe that different text phrases (highlighted in bold) for the named entity ‘Taylor Swift’¹¹ have been used in different tweets. End users adapt different text phrases based on their popularity, the current trends or simply due to limited character space. While many of the text phrases which are used frequently in such texts may be indexed by the KB for the named entity (such as shown above in the examples), many of the unpopular or infrequently used ones may not be indexed. Such phrases can be defined as *Out of Vocabulary Mentions*.
3. **NIL Mentions**: This category is represented by entity mentions that have been either correctly identified, however, the mentions may not be popular enough for describing the corresponding named entities in the KB, and thus a suitable KB resource match is not found; or the mentions have been incorrectly identified by an entity recognition system, i.e., the said mention does not denote a named entity in the real world and, thus, cannot be linked to any KB resource. Such mentions have been denoted as *Out of Knowledge Base* or *Unlinkable Mentions* in this thesis. Further insights towards NIL Mentions is provided in Section 2.3.3.1.

¹¹http://dbpedia.org/page/Taylor_Swift

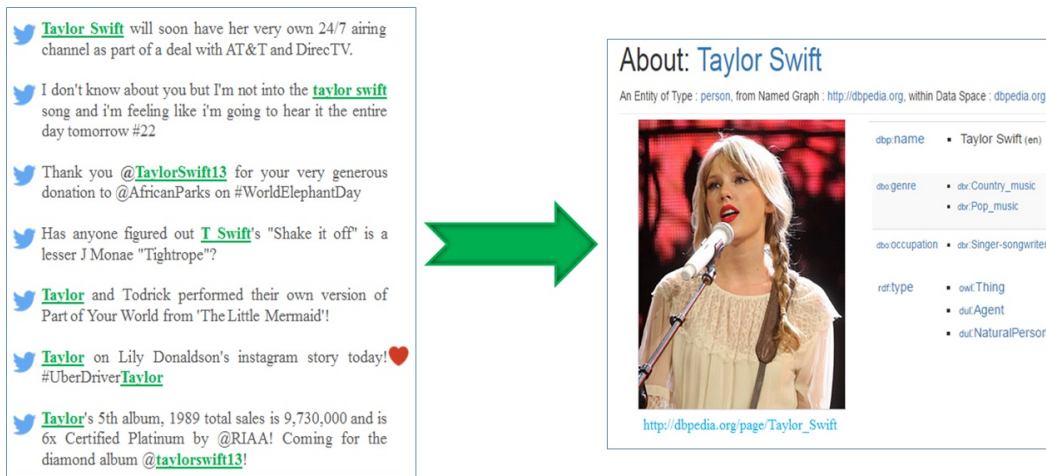


Figure 2.4: Example: Synonymous Entity Mentions

An entity mention that exhibits ambiguity in any of the above described forms poses difficulties for an entity recognition system and, therefore, for a linking system in a NEEL framework. In order to identify and classify entity mentions as well as to deal with ambiguity, several state-of-the-art models [74, 102, 134] leverage different kinds of contextual information, as seen in Table 2.1, available in the analyzed text (e.g., the use of articles and/or prepositions surrounding the entity mention), as well as in other corpora (e.g., the distribution of words across different classes of entity mentions represented in a knowledge base).

Problem 1: While contextual information that can be used by Natural Language Processing (NLP) methods is quite reliable and abundant in well-formed texts (yielding to good accuracy of entity classifications methods), it is less reliable and scarcer in microblogging environments [31]. As a result, entity identification and classification is often inaccurate in this context; specifically when it comes to ambiguous entity mentions; which consequently effects the linking performance in NEEL frameworks.

*An unsupervised approach named **Linking2Adapt** has been proposed, to this end, and presented in Chapter 3, Section 3.2 and a detailed investigation has been performed, as a means to tackle this problem, with regards to the usage of*

contextual evidence from a tweet (i.e., evidence from the Unstructured Web) as well as from a KB resource (i.e., evidence from the Semantic Web) so as to be able to improve entity identification and classification in a proposed NEEL framework.

2.3.2 Ontologies for Named Entity Classification

As described above, the task of Named Entity Classification deals with the categorization of identified entity mentions into a set of pre-defined (mostly disjoint) entity types (or classes), such as PERSON, LOCATION, and ORGANIZATION according to an underlying ontology. The classification performance of a NER system is dependent on several factors as listed below:

- text segmentation, where segmentation errors can lead to identification of a non-mention as a mention associated with a specific entity type;
- identification performance, where identification errors percolating from the entity identification phase can lead to classification of incorrectly identified non-mentions into entity types;
- presence of a polysemous mention with not enough context to be able to distinguish between a possible set of classes for the mention;
- the ontology classes (as discussed below) are too generic to be able to deal with an entity mention belonging to a particular class with regards to its class in a given KB;
- not enough training data; and
- an *Out of Knowledge Base* mention (or in other words, a newly evolved mention).

Given these scenarios and the diversity of information available today from different platforms, it is often the case that new and different ontologies and/or existing ontologies with refined levels of granularity have to be adopted so as to classify these diverse forms of information. Another important factor is that new information is always evolving in real-time, as

mentioned above, and subsequently, such information will have their own inherent semantic structure.

The term *Ontology*, also described previously, is used to denote a classification hierarchy or a taxonomy for a particular domain with classes or concepts according to the domain in consideration. Knowledge bases, in particular, DBpedia is known to have a complex, detailed and a cross-domain ontology with 685 classes, described by 2,795 properties. Classes in DBpedia are described by properties. For instance, the class *Person* is described by properties such as *birthName*, *age*, *bloodGroup* etc. Entity recognition systems, on the other hand, use rather simplistic ontologies or classification taxonomies. For instance, the MUC taxonomy consists of the classes PERSON, LOCATION, ORGANIZATION, DATE, TIME, MONEY, and PERCENT [51]. State-of-the-art NER systems such as ANNIE [26] are based on such taxonomies, while the Stanford Named Entity Recognizer [43] uses 3, 4 or 7 different classes, as described below, depending upon a user's requirements:

1. *3 class model*: consists of the classes LOCATION, PERSON, and ORGANIZATION based on CoNLL 2003, MUC 6, MUC 7, and ACE 2002 datasets [51, 123];
2. *4 class model*: consists of the classes LOCATION, PERSON, ORGANIZATION, and MISC based on CoNLL 2003 task datasets [123]; and
3. *7 class model*: consists of the classes LOCATION, PERSON, ORGANIZATION, MONEY, PERCENT, DATE, and TIME based on the MUC 6 and MUC 7 training data sets [51].

T-NER, another state-of-the-art NER system [102], uses the classes BAND, COMPANY, FACILITY, GEO-LOCATION, MOVIE, OTHER, PERSON, PRODUCT, SPORTSTEAM and TVSHOW for classification of identified entity mentions. Depending upon application requirements and the domain in consideration, different NER systems are, thus, used with different ontologies.

This can however lead to difficulties in performance evaluations as well as integration of various NER systems. In such cases, a system using one

(source) ontology may need to be adapted to one using a different (target) ontology according to application requirements. For example, in a domain with strong focus on **music**, one may want to use distinct classes for *music artists* and *bands*, while in another domain more centered on **movie**, one may want to distinguish between *persons* and *fictional characters*. An adaptation, in this context, would essentially mean developing a mapping or an alignment [2, 114] between classes of the source and target ontologies in consideration. This mapping is important not only from the point of view of being able to compare the performances of different NER systems, but also from the point of view of accomplishing interoperability among the given ontologies on the Semantic Web, as well as bringing about robustness to the process of entity classification so as to be able to deal with constantly evolving, and difficult to classify fresh information. To this end, most practitioners establish mappings between concepts of ontologies in manual ways. As a matter of fact, an important investigation from this perspective has been presented in [106], where a web-based application named NERD (Named Entity Recognition and Disambiguation) is built on top of numerous NER classifiers such as AlchemyAPI¹², DBpedia Spotlight, Zemanta¹³, Extractiv¹⁴, OpenCalais¹⁵ and so on. The NERD API is supported by its own *NERD ontology*, which is established by manually mapping the ontology concepts of the NER classifiers with the concepts of NERD.

It has been observed that when many-to-one (m:1) mappings are used, i.e., when one source concept is mapped to at most one target concept, and when the source classification is reasonably accurate, manual mappings may achieve a reasonable performance. However, in contexts such as microblogging platforms, where gross-grained ontologies are used for classification, these mappings have several limitations as discussed in Section 3.3.2. Further, transfer learning approaches have been used for mapping the concepts of ontologies of NER classifiers [1, 29] for formal texts. In particular, the approach proposed in [1] learns a domain-independent base model, which

¹²<http://www.alchemyapi.com/>

¹³<http://www.zemanta.com/>

¹⁴<http://extractiv.com/>

¹⁵<http://www.opencalais.com/>

can be adapted to specific domains. On the other hand, domain adaptation approaches [53] have been proposed as well for this task. In [108], the authors investigate ontology alignment of NER classifiers of different domains by automatically inducing mappings between concepts of different ontologies, as opposed to their previous approach [106] based on the use of manual mappings.

Additionally, from the point of view of interoperability as well as integration of the heterogeneous data available on the Web, there has been consistent work towards ontology mapping, matching and alignment in the Semantic Web community [32,35,118]. To this end, simplistic similarity measures such as edit distance, Jaro-Winkler distance, and linguistic measures [127] as well as complicated measures such as joint probability distribution of ontology concepts [32] have been used to estimate similarities between concepts (or in this case, entity types) of different ontologies. However, when it comes to the task of mapping entity types of NER classifiers, a mere string similarity measure or a hand-crafted rule for manual mapping may not suffice. The identified entity mentions and their contextual usage (among other factors) also need to be considered for a more robust mapping. For instance, in many cases, a polysemous mention such as *Harry Potter* might be mapped from an entity type, say, *Character* in a source ontology to *Person* in the target ontology, or from the type *Film* in a source ontology to *Product* in the target ontology.

Problem 2: In the afore-mentioned scenarios, mapping across ontologies used by different systems or requiring a system that uses a source ontology to classify mentions using a different target ontology is not trivial. Essentially, such tasks in state-of-the-art systems have been performed manually [106], i.e. mappings between ontology classes have been developed manually based on an expert intuition, as mentioned above. However, such mappings are important when ontologies of cross-domain NER systems, or KBs with highly complex, fine-grained ontologies are under consideration and, thus, there is a need of automation of this task.

*A supervised approach named **Learning2Adapt** has been proposed (inspired by the transfer-learning paradigm) in this thesis, and presented in Chapter 3, Section 3.3, for the purpose of automatically learning a mapping from an entity type of a given source ontology to an entity type of the target ontology by exploring factors such as contextual usage of an entity mention in a tweet.*

The approach proposed in [108] is one of the principle approaches for Learning2Adapt, in the sense, that the authors advocate that mappings between ontology concepts can be contextual, by inductively learning mapping between ontology concepts. Learning2Adapt differs from that in [108], in the sense that probabilistic distribution over concepts of a source ontology for an entity mention has been used to learn the corresponding concept in the target ontology by a transfer learning (domain-adaptation) approach. By considering a probabilistic distribution, the mapping becomes dependent on the context of the mention (i.e., instance-based mappings). In other words, Learning2Adapt addresses the ontology mapping problem from the perspective of an instance’s (entity mention) usage, which in turn deals with the problem of polysemous entity mentions found in texts. This would mean that a polysemous mention would have different probabilistic distributions over the concepts (or entity types) depending upon its usage contexts, and hence, would be mapped to different entity types in the target domain, as opposed to the use of handcrafted rules where usage contexts are mostly not taken into consideration. To the best of our knowledge, previous work has not addressed the problem of automatically adapting the ontology used by a NER system from this perspective, so as to comply to a new ontology.

2.3.3 Named Entity Linking

The task of grounding or linking a given entity mention to an appropriate resource in a knowledge base, which is representative of the named entity in the real world, is referred to as **Named Entity Linking**. Figure 2.5 shows entity linking from two textual formats, i.e., newswire text (long text) and tweets (short text) in the simplest sense. While the mentions $\{Ben\ Carson, Carson\}$, $\{Donald\ Trump, president-elect\}$

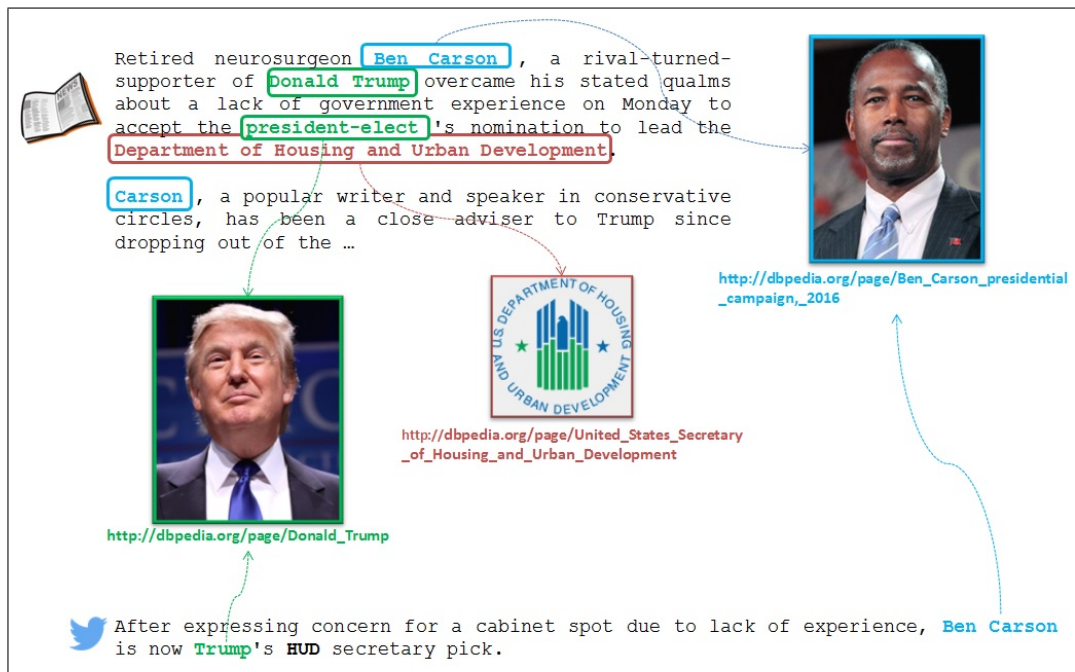


Figure 2.5: Named Entity Linking: NewsWire Text vs Tweets

and $\{Department\ of\ Housing\ and\ Urban\ Development\}$ identified from the newswire text are easily linked to their counterparts in DBpedia (as indicated by the URIs), the linking of the mention *HUD* found in the tweet to the KB resource for *Department of Housing and Urban Development* can be apparently difficult due to lack of context.

2.3.3.1 Candidate Match Retrieval and Disambiguation

Conventionally, for being able to link an entity mention, a candidate look-up is performed from a local index or a KB and a list of, say, top-k candidate resources is retrieved for the given mention. Further, the mention is disambiguated with the most appropriate candidate resource based on account of factors such as lexical similarity between the entity mention and a candidate resource. However, in many cases, a candidate look-up process may not return a list of candidate resources. This can happen in the following two cases:

1. The entity mention is an *Out of Vocabulary* (OOV) mention, which

may mean that the named entity has been referred to by a mention which has not been indexed (yet) in the KB for the said named entity. In other words, as described in Section 2.3.1, it can be understood by the case of *synonymy* where a named entity can be denoted by multiple entity mentions in an unstructured text. In this case, the synonymous mention for which candidate look-up is being performed has not been used in the KB as a means to describe the named entity, due to reasons such as it being an unpopular mention or a new mention, although the named entity does exist in the KB. For instance, the entity mention *HUD* in the tweet may be considered as an OOV mention, since the named entity pertaining to the ‘Department of Housing and Urban Development’ exists in the KB (as indicated by the URI), however, the entity mention *HUD* does not. Hence, no relevant candidates are returned by candidate look-up.

2. The entity mention is an *Out of Knowledge Base* (OOKB) (also known as unlinkable) mention, which may mean that neither the named entity nor the entity mention exist in the KB. This can be attributed to the mention being either an incorrectly identified mention (thus a suitable match cannot be found) or the mention refers to a new named entity in the real world which has recently evolved and has not been indexed yet (or may not be relevant/popular enough to be indexed).

Many entity linking systems have been proposed in the state-of-the-art [44, 57, 63, 84, 98, 99] for linking mentions recognized from different textual formats and from multiple languages, while using different KBs (such as Wikipedia, DBpedia or domain-specific KBs) for candidate look-up and entity disambiguation. While long textual formats can provide apt background information into, say, an OOV mentions’s contextual usage, short textual formats like microblogs are especially difficult to deal with on the other hand. An overview of state-of-the-art entity linking approaches for short texts (like tweets) is presented below.

An inferencing mechanism is used in [73] for linking tweet entity mentions to candidate KB resources which takes into consideration similarity between entity mentions, similarity between an entity mention and a candidate KB re-

source, and similarity between candidate KB resources of an entity mention. In [113], a graph-based framework is proposed for entity linking in tweets by modelling a user’s interests. By doing so, they deal with the absence of sufficient contextual information. In [62], an entity linking framework is proposed for linking entities in tweets by leveraging contextual information, including entity popularity and entity recency which model users’ evolving interest patterns in the real world. Users’ interests are modelled by using the social interaction of a user with another, which can, thus, capture diverse user interests. Furthermore, an approach to improve tweet entity disambiguation is proposed in [50], where a tweet’s content is enriched by using a) hashtag definitions harvested from the Web (in the cases, where a hashtag is used in a tweet), b) Twitter profile information for a @username (in the cases, where a username is present in a tweet), c) web content information for URLs (in the cases, where a URL has been used in a tweet), and d) a combination of all these information. A machine learning based approach is proposed in [80] using n-gram features, concept features, and tweet features in order to identify concepts that are semantically related to a tweet, thereafter, generating links to Wikipedia articles for every entity mention in a tweet. Finally, an entity linking approach for tweets is proposed in [131] where candidate entities in a tweet are identified using n-gram and string matching methods, followed by use of *random forest* method to establish correct links for the entities. This information is further used in order to improve the task of entity recognition. Other details for these and several other approaches w.r.t. the taxonomies and the datasets used for experimental analysis is presented in Table 2.2.

Lately, the orchestration of entity recognition, disambiguation and linking components in the form of NEEL frameworks specifically for microblog posts has also opened up interesting research directions in terms of identifying existing knowledge and discovering new knowledge from microblogging platforms. An important state-of-the-art graph-based approach named Babelfy¹⁶ has been proposed in [84] where entity linking and word sense disambiguation are combined in order to identify and disambiguate entity mentions (from any textual format) and link them using a multilingual semantic network named BabelNet [89]. This approach integrates the lexicographic knowledge used

¹⁶<http://babelfy.org>

Table 2.2: State-of-the-art NEL approaches

S.No.	Name of the Approach	Learning Methods	Features	Datasets	Domain	Referenced KB
1	TAGME Ferragina et al., 2010 [40]	Anchor Disambiguation & Pruning	Use of Wikipedia anchor texts & pages linked to those texts	<ul style="list-style-type: none"> IITB dataset [68] Wikipedia datasets 	Short textual formats (news, tweets etc.)	Wikipedia
2	Meij et al., 2012 [80]	Semantic Linking using supervised learning	n-gram features, concept features, and tweet features	Manually annotated dataset	Twitter	Wikipedia
3	Babelify Moro et al., 2014 [84]	Graph-based approach	Random Walk with Restart for creating semantic signatures for each concept & named entity in the BabelNet semantic network	<ul style="list-style-type: none"> SemEval-2013 task 12 dataset [87] SemEval-2007 task 7 dataset [88] SemEval-2007 task 17 dataset [95] Senseval-3 dataset [116] KORE50 [60] AIDA-CoNLL [61] 	Short text	BabelNet semantic network
4	Gorrellet al., 2015 [50]	YODIE for entity recognition and disambiguation	Use of additional tweet context for entity disambiguation (hashtags, user profile, URLs)	Manually annotated dataset	Twitter	DBpedia
5	S-MART Yang et al., 2015 [132]	Structural Learning (Tree-based)	<ul style="list-style-type: none"> Generalize multiple additive regression Trees for structured learning N-gram matching 	<ul style="list-style-type: none"> #Microposts2014 [8] Fang and Chang 2014 [38] 	Twitter	Wikipedia

in word sense disambiguation and encyclopaedic knowledge used in entity linking in a complimentary fashion where one knowledge helps to tackle the other task. Further, NERD (Named Entity Recognition and Disambiguation) has been proposed in [105] which is built on top of various popular named entity extractors (viz, AlchemyAPI, DBpedia Spotlight, Extractiv, OpenCalais and Zemanta) as a means to evaluate the entity extraction and disambiguation performances of these systems from any textual format. The use of Wikipedia has been proposed as a KB resource and a contextual reference for a named entity recognition and disambiguation system in [25], where entities are extracted from *titles of entity pages, titles of redirecting pages, disambiguation pages, and references to entity pages in other Wikipedia articles* using a hybrid named entity recognizer (from the English Reuters corpus available from CoNLL 2003 [123]) and disambiguation is performed by using a vector space model where Wikipedia category information, and contextual information of an entity (obtained from an entity’s Wikipedia page) is augmented with information extracted from Wikipedia list pages to form a vector representation and compared to vector representations of Wikipedia entities.

An approach for entity detection (or recognition) and disambiguation has been proposed in [54] as a means to establish an end-to-end entity linking pipeline for tweets using Structural Learning, where candidate entity mentions are identified using a conventional *k-grams* approach and entity linking is performed using several features such as popularity of an entity in a KB, capitalization features, Tf-idf scores and type of the entity in the KB. Another end-to-end entity linking approach is proposed in [130] for tweets, where entity mentions are identified in a tweet by exploiting a dictionary derived by Wikipedia. Various features such as contextual information from a tweet, temporal entity popularity, and string similarity measures are taken into account in order to generate candidate lists for an entity mention. Once an entity mention has been found, and its possible referent candidate resources have been identified by string matching with Wikipedia resources, a random forest is adopted to learn the patterns underlying the correct linking. Mentions that do not have a corresponding KB match are categorized as NIL Mentions. The prediction provided by random forest are then exploited in a

further random forest that, together with other features such as number of in-bound links and average page view, predicts the entity type.

Further, an industrial end-to-end entity recognition and linking system for tweets has been proposed in [47] using a global, real-time KB built using Wikipedia, where entity recognition is performed by parsing a tweet to extract strings that match the instances in the KB, in addition to the use of off-the-shelf named entity recognizers to improve recognition efficiency. The mentions are then disambiguated with nodes/instances in the KB based on disambiguation scores, which are based on features such as node popularity in the KB and similarity scores between a mention and its usage context in the tweet with the Web context of the node in the KB. In [11], an approach for recognizing entities by generating candidate resources for all tokens tagged as proper nouns (using a PoS tagger) in a tweet has been proposed, followed by disambiguating and linking the recognized entities using an adaptation of the distributional Lesk algorithm proposed in [10]. Another end-to-end entity linking framework for tweets is proposed in [20], where entity mentions are generated using a lexicon-based approach, wherein a dictionary is used as a lexicon (constructed from Wikipedia and Freebase) to extract entity mentions from tweets which are further associated with a set of candidate resources (to be linked to), followed by entity disambiguation and linking based on a supervised learning approach. Features such as textual usage of an entity mention in a tweet (in terms of its context), graphical features which capture the semantic cohesiveness between an entity mention and a candidate resource and statistical features which capture a candidate’s usage popularity in the lexicon are used for training the learning model to disambiguate and link an entity mention with an appropriate candidate resource. Other details for these and several other approaches w.r.t. the taxonomies and the datasets used for experimental analysis is presented in Table 2.3.

An important observation to draw here is that not only very few NEEL frameworks exist for recognizing as well as linking entity mentions from tweets to knowledge bases, but also most of them are quite recent. Further, many frameworks as well as independent linking approaches have seen to be using pre-existing tools or oracles for entity recognition to generate a candidate list of entity mentions for addressing the task of entity linking.

Table 2.3: State-of-the-art NEEL approaches and frameworks

S.No.	Name of the Approach	Learning Methods	Features	Datasets	Domain	Referenced KB
1	Gattani et al, 2013 [47]		<ul style="list-style-type: none"> Dictionary-based approach for NER, use of off-the-shelf NER systems Disambiguation based on candidate popularity, similarity scores 	<ul style="list-style-type: none"> Manually annotated datasets 	Twitter	Wikipedia
2	Guo et al, 2013 [54]	Structural SVM (Mention detection & disambiguation)	<ul style="list-style-type: none"> First-Order features: Mention-specific features (entity mention & tweet), Entity-specific features (KB content for a candidate), Candidate popularity, Candidate type Second Order Features like Jaccard distance 	<ul style="list-style-type: none"> Manually annotated datasets Ritter et al, 2011 [102] 	Twitter	Wikipedia
3	Chang et al, 2014 [20]	Supervised Learning	<ul style="list-style-type: none"> Use of lexicon to generate entity mentions Use of textual features, statistical features & graph-based features for NEL 	#Microposts2014 [8]	Twitter	Wikipedia & Freebase
4	NEED4TWEET Habib et al, 2015 [56]	<ul style="list-style-type: none"> CRF for NER SVM for candidate ranking (NED) 	<ul style="list-style-type: none"> Tweet Segmentation & n-gram lookup for NER, Use of contextual & semantic features for disambiguation 	<ul style="list-style-type: none"> Locke and Martin [75] Habib and van Keulen [55] #Microposts2014 [8] 	Twitter	Wikipedia & YAGO
5	Basile et al, 2015 [11]	<ul style="list-style-type: none"> Supervised & unsupervised approaches for NER Dist. Lesk algorithm for NEL 	<ul style="list-style-type: none"> Supervised (use of PoS-tag information for NER) Unsupervised (use of n-grams to search Wikipedia titles) Vector similarity between context for an entity mention (retrieved from a tweet) & glosses (extended abstracts from DBpedia) associated with all candidates for a mention 	#Microposts2015 [104]	Twitter	DBpedia
6	Yamada et al, 2015 [130]	<ul style="list-style-type: none"> Use of dictionaries & n-grams for NER Random Forest for scoring & linking entity mentions 	<ul style="list-style-type: none"> Use of word embeddings for computing contextual similarity between a tweet & a KB resource String similarity between an entity mention & KB resource Temporal popularity of a resource in a KB 	#Microposts2015 [104]	Twitter	DBpedia
7	Cucerzan et al, 2007 [25]		<ul style="list-style-type: none"> Use of Wikipedia statistics & Web search results for NER Use of vector space model for disambiguation by making use of contextual & category information extracted from Wikipedia 	<ul style="list-style-type: none"> News Articles Wikipedia Articles 	Long textual format	Wikipedia
8	Damjanovic et al, 2012 [28]	<ul style="list-style-type: none"> NER using ANNIE [27] NEL using GATE's ontology-based gazetteer [15] 	Disambiguation scores computed using string similarity, structural similarity & contextual similarity.	Manually annotated dataset of 100 Wikipedia user profiles	Long textual format	DBpedia

This shows that there is a lot of scope for orchestrating the recognition and linking tasks towards a functional entity recognition and linking framework. Another important observation is that, while frameworks exist for recognition and linking tasks, no major steps (except [131]) have been taken towards performance optimization by utilizing one component of the framework for the other.

Problem 3: Poor performance of one component eventually effects the performance of other components in a NEEL framework. For microblogging textual formats, where there is always an inherent lack of context and presence of noise, a sub-par performance of any component will have a major influence on the overall performance of the NEEL framework. It is, thus, important to build a framework whose components can be adapted gradually in a way so as to optimize the overall efficiency of the framework.

*One of the major contributions of this thesis is the orchestration of recognition and linking components towards a NEEL framework, which is then utilized to investigate the effect of one component on the performance of the other and the overall framework. To this end, an unsupervised approach named **Linking2Adapt** (Chapter 3, Section 3.2) has been proposed with a goal to adapt entity identification and classification performed by a recognition component of the framework based on the linking component. Further, a supervised approach named **Learning2Link** (Chapter 3, Section 3.4) has been proposed with a goal to adapt the linking component of a NEEL framework based on the recognition and disambiguation components of the framework.*

The challenges and tasks of entity recognition and linking have been described in this chapter. Further, the motivating factors for the contributions of this thesis have also been highlighted. A detailed account of the approaches proposed and their mathematical representations is presented in the next chapter.

3 A NEEL FRAMEWORK

A Named Entity Extraction and Linking (NEEL) Framework, as mentioned in Chapter 2 Section 2.3, constitutes an entity recognition component, an entity disambiguation component and an entity linking component. Additionally, a pre-processing component has been used by many state-of-the-art frameworks for filtering out noise and parsing the text into segments. These components were described briefly in Chapter 2. Today, such frameworks are beneficial for tasks such as product recommendations, semantic searches and knowledge base enrichment. However, due to the challenges posed by microblogging platforms, limited NEEL frameworks exist for such platforms, most of which are for commercial purposes only. One of the main goals of this thesis has been to define an advanced NEEL framework, in particular for microblogging platforms, which makes use of multiple evidences obtained from the components of the framework (as shown below in Figure 3.1), and investigating the role of one component for performance improvement of another as well as of the whole framework.

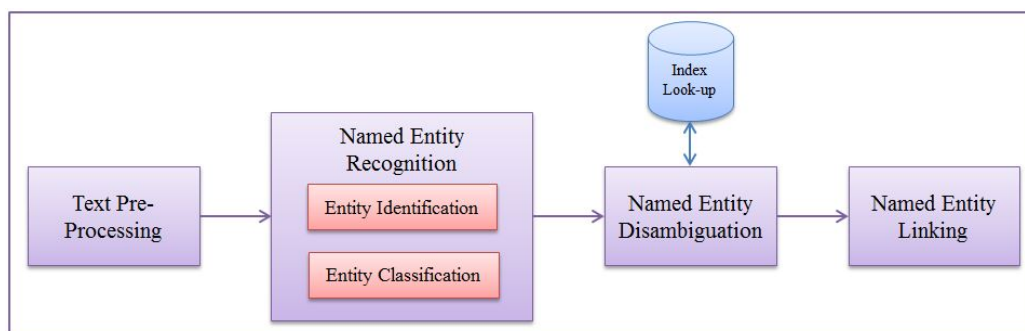


Figure 3.1: NEEL Framework

3.1 Overview of the Proposed Framework

This section presents an overview of the NEEL framework that has been proposed for microblogging platforms, in particular Twitter.

Named Entity Recognition (NER): As described in Section 2.3, a recognition component of a NEEL framework is responsible for identifying entity mentions from a given text, and classifying them into entity types according to a given ontology (O_S). For the task of identifying entity mentions from text (in this case, tweets), Conditional Random Fields (CRF) [69] has been used, which is a probabilistic undirected graphical model used for segmenting and annotating the text. Based on this model, every identified entity mention e_j can belong to one or more entity types/classes $c(e_j)$ of an ontology, each associated with a probability denoted as $P_{CRF}(e_j, c(e_j))$, which denotes the probability that an entity mention e_j belongs to an entity type/class $c(e_j)$. $P_{CRF}(e_j, c(e_j))$ is defined as follows:

$$P_{CRF}(e_j, c(e_j)) = \exp\left(\sum_{k=1}^K w_k f_k(e_j, c(e_j))\right) \quad (3.1)$$

where w_k are the weights learned from data and f_k are the feature functions encoded by CRF. The output of CRF is a set of entity mentions e_1, e_2, \dots, e_m identified from a given tweet t . This model (shown as NER (1) in Figure 3.2) has been trained according to a given ontology O_S . With the help of the proposed framework, this model can be adapted to any new ontology O_T to derive a new model NER (2). More details regarding the need to adapt a NER model based on one ontology to another is provided in Section 3.3.

Named Entity Disambiguation (NED): After entity recognition, a candidate resource selection and ranking step is executed in order to aid the entity disambiguation step. This means that, for every identified entity mention, a suitable list of candidate resources is retrieved from a KB so as to disambiguate it with the most suitable candidate resource in the KB. For this purpose, the DBpedia knowledge base has been used. To this end, all

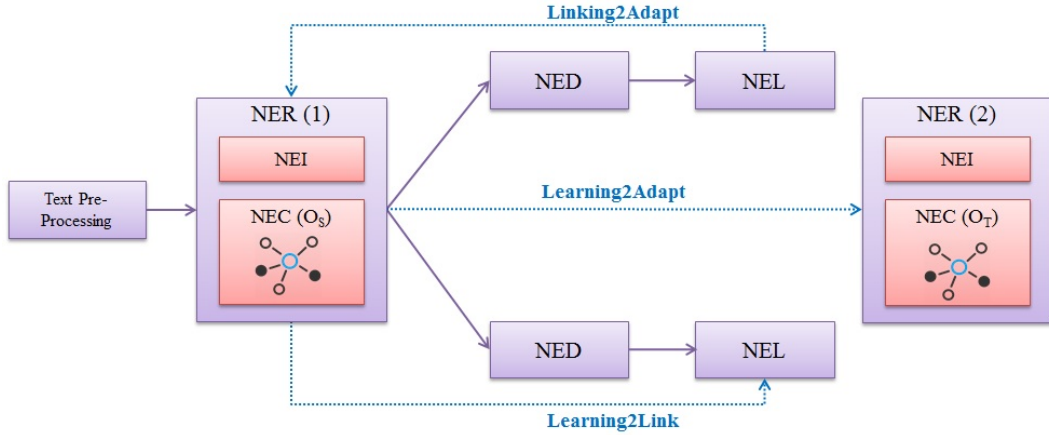


Figure 3.2: Proposed Approaches

the ‘Titles’ of all *Wikipedia articles*¹ from DBpedia using *rdfs:label* have been extracted and indexed locally using LuceneAPI². Next, for each mention, top-k candidate KB resources have been retrieved from the index using a high-recall approach and a knowledge-base score denoted by $KB(e_j, r_k)$ is estimated, for each candidate resource r_k of an entity mention e_j . The mathematical definition of $KB(e_j, r_k)$ as well as the features considered for estimating a knowledge-base score are provided in the description of the proposed approaches. The top-k candidate resources which have been retrieved are ranked based on their respective (normalized) knowledge-base scores.

Named Entity Linking (NEL): In this component, an entity mention is linked to the most suitable candidate resource (if any candidate resources are retrieved for the given mention), which is decided based on the KB score $KB(e_j, r_k)$ obtained in the previous step.

The proposed framework has been used to realize the approaches which have been proposed in this thesis (as also listed in Chapter 1) in order to improve the performance of the framework. They are summarized below and shown in Figure 3.2:

¹<http://dbpedia.org/Downloads2015-04>

²<http://lucene.apache.org/>

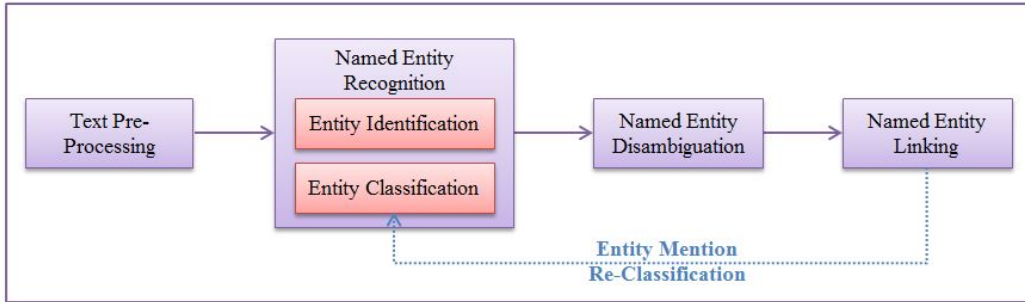
1. **Linking2Adapt:** This approach has been proposed to adapt NEI (by means of an unsupervised *entity mention re-scoping* approach) and NEC (by means of an unsupervised *entity mention re-classification* approach) performed by the recognition component by using the linking component in the NEEL framework. This approach is discussed in detail in Section 3.2.
2. **Learning2Adapt:** This supervised approach has been proposed to adapt NEC performed by a recognition component in a NEEL framework for different ontologies (used by different named entity classifiers). This approach is discussed in detail in Section 3.3.
3. **Learning2Link:** This supervised approach has been proposed to adapt the linking component by using the recognition and disambiguation components in a NEEL framework. It has been discussed in detail in Section 3.4.

3.2 Linking2Adapt

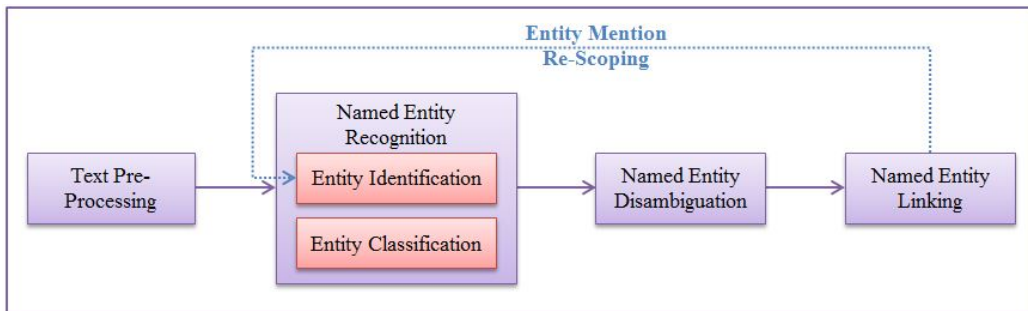
Linking2Adapt is an unsupervised approach, which has been proposed in order to investigate the effect of linking on entity classification (NEC) by means of an *entity mention re-classification* technique. Further, Linking2Adapt also investigates the effect of linking on entity identification (NEI) by means of an *entity mention re-scoping* technique. Both of these methods are described in Figure 3.3 below.

3.2.1 Entity Mention Re-Classification

As stated above, this is an unsupervised approach, which has been proposed with the intent of improving the performance of NEC by a feedback technique to *re-classify* the identified entity mentions based on the insights that can be obtained by a linking phase. A graphical representation of this approach is shown in Figure 3.3(a).



(a) Linking2Adapt: Entity Mention Re-Classification



(b) Linking2Adapt: Entity Mention Re-Scoping

Figure 3.3: Linking2Adapt

Named Entity Recognition: Firstly, before proceeding with the tasks of entity recognition and linking, text (in this case, tweet) pre-processing is performed wherein special characters (such as @, #,..) are removed. Further, for entity recognition, a state-of-the-art NER system called T-NER [102] has been used, which is a pre-trained system on a state-of-the-art gold standard of Twitter data using an underlying ontology, O_S (known as Ritter Ontology) to finally derive a NER model R_S . It is a supervised model based on Conditional Random Fields (CRF) [119] and performs segmentation of a tweet while subsequently classifying each token according to the classes of its ontology, as reported below:

Ritter (T-NER) Ontology (O_S): *Band, Company, Facility, Movie, Geo-Location, Organization, Other, Person, Product, Sportsteam, TVshow.*

Further, T-NER uses IOB encoding, as described in Section 2.3 (i.e., each word is either inside/outside/beginning of an entity mention), for named entity segmentation for identifying the segments as mentions or non-mentions and further, classifying the mentions based on its ontology. Thus, every identified mention e_j is classified into one or more entity types $c(e_j)$ (belonging to O_S) with a probability denoted by $P_{CRF}(e_j, c(e_j))$ indicated in equation (3.1). This probability represents an *a priori* estimation of the entity type that will be exploited in the subsequent phases. Concerning the time complexity, this step requires $\mathcal{O}(T \times |C|^2)$ for each tweet, where T denotes the length of the tweet and $|C|$ is the number of entity types.

Named Entity Disambiguation: For every identified entity mention, a list of top-k candidate resources is retrieved from the local index using a high-recall approach, as also discussed in Section 3.1, where an empirically defined knowledge-base score denoted by $KB(e_j, r_k)$ is estimated for each candidate resource r_k of an entity mention e_j as follows:

$$KB(e_j, r_k) = (\alpha \cdot lex(e_j, l_{r_k}) + (1 - \alpha) \cdot (cos_k(e_j^*, a_{r_k}))) + R(r_k) \quad (3.2)$$

where:

- $lex(e_j, l_{r_k})$ denotes a lexical similarity between an entity mention e_j and the label of a candidate resource l_{r_k} ;
- $cos_k(e_j^*, a_{r_k})$ represents a discounted cosine similarity between an entity mention’s context e_j^* and a candidate resource’s KB abstract description a_{r_k} ;
- $R(r_k)$ is a popularity measure of a given candidate in the KB.

Here lexical similarity $lex(e_j, l_{r_k})$ is defined as follows:

$$lex(e_j, l_{r_k}) = lcs(e_j, l_{r_k}) + W_D \left(\frac{JW(e_j, l_{r_k})}{W_D + 1} \right) \quad (3.3)$$

where $lcs(e_j, l_{r_k})$ denotes a normalized Lucene Conceptual Score³ between e_j and l_{r_k} and is estimated in order to filter out false positives obtained due

³https://lucene.apache.org/core/4_6_0/core/org/apache/lucene/search/similarities/TFIDFSimilarity.html

to high recall, while $W_D \left(\frac{JW(e_j, l_{r_k})}{W_D + 1} \right)$ represents a string distance measure, based on the well-known *Jaro-Winkler distance*, between an entity mention and the label of a candidate resource. The coefficient W_D is set equal to 3.0 and represents a boosting coefficient that allows to weigh close matches more syntactically. It can be observed from Equation (3.3), when W_D is replaced with its coefficient value (i.e., $W_D = 3.0$), that the Jaro-Winkler distance has been assigned an empirically defined relative weight of $(0.75 \cdot JW(e_j, l_{r_k}))$ while $lcs(e_j, l_{r_k})$ has been assigned a relative weight of $(1.0 \cdot lcs(e_j, l_{r_k}))$. This has been done so as to give more weightage to the Lucene score as compared to the Jaro-Winkler distance between an entity mention and a label of a candidate resource. Further, the asymmetric Jaro-Winkler distance weighs more edit distances occurring in the first sub-sequences of two strings, and is defined as:

$$JW(e_j, l_{r_k}) = Jaro(e_j, l_{r_k}) + \frac{P'}{10} \cdot (1 - Jaro(e_j, l_{r_k})) \quad (3.4)$$

where *Jaro* is a similarity metric [64] and P' is a measure that takes into account the length of the longest common prefix of e_j and l_{r_k} . More formally, the Jaro similarity between two strings s and t is defined as follows:

$$Jaro(s, t) = \frac{1}{3} \cdot \left(\frac{|s'|}{|s|} + \frac{|t'|}{|t|} + \frac{|s'| - T_{s',t'}}{|s'|} \right) \quad (3.5)$$

Moreover, in situations where a candidate label l_{r_k} is composed of more than one token, $JW(e_j, l_{r_k})$ is calculated as follows:

$$JW(e_j, l_{r_k}) = \max(JW(e_j, P_1^{l_{r_k}}), \dots, JW(e_j, P_n^{l_{r_k}})) \quad (3.6)$$

where $P_i^{l_{r_k}}$ denotes one of every possible permutation of tokens in l_{r_k} . This particular step is undertaken because end users might refer to a named entity in a tweet using concise, popular substrings (entity mentions) of the named entity, which may not necessarily be the first token of the entity itself. For instance, in the tweet,

```
@steph93065 shes hates me but she's no bigot,
intelligent and correct most of the time. #Trump
```

it is observed that candidate KB resources for the entity mention ‘Trump’ comprise of *Trump* (card game, rdf:type Thing), *Donald_Trump* (rdf:type Person), and *Trump_(comics)* (rdf:type CartoonCharacter), amongst other candidates. By using the afore-mentioned equation (3.6), the JW distance for the entity mention ‘Trump’ was computed not only with ‘Donald Trump’, which yields a low JW similarity, but also with ‘Trump’, which yields a high JW similarity.

The second component $\text{cos}_k(e_j^*, a_{r_k})$ is evaluated by making use of a Vector Space Model representation [23]. In particular, the contextual information of a mention e_j , denoted as e_j^* , comprises of the entity type $c(e_j)$ derived by the T-NER system, the surrounding content (i.e. nouns/verbs/adjectives) available in the tweet and the mention itself. In order to evaluate the second component $\text{cos}_k(e_j^*, a_{r_k})$ of the KB score in equation (3.2), the *long/full abstracts* (dbo:abstract) of all DBpedia resources have been indexed. This has been done with an objective to be able to disambiguate an entity mention with a candidate label using a mention’s usage context in the tweet, on one hand, and contextual evidence from the KB on the other. The measure $\text{cos}_k(e_j^*, a_{r_k})$, which is used for denoting contextual similarity between an entity e_j and a KB candidate resource r_k , is defined as:

$$\text{cos}_k(e_j^*, a_{r_k}) = \begin{cases} \text{cos}(e_j^*, a_{r_k}) & \text{if } k = 1 \\ \frac{\text{cos}(e_j^*, a_{r_k})}{\log_2(k)} & k \geq 2 \end{cases} \quad (3.7)$$

To compute equation (3.7), the abstracts for all the top-k candidate resources r_1, r_2, \dots, r_k are retrieved from the index. Equation (3.7) allows the system to scale the similarity with respect to each candidate abstract according to its ranking position. Finally, the last component provided in equation (3.2) is provided by $R(r_k)$, which allows the system to take into account the popularity of a given candidate resource in the KB so as to be able to improve the candidate selection and the subsequent ranking. For this purpose, the popularity $R(r_k)$ of a KB resource r_k is computed by using the following boosted Page Rank coefficient:

$$R(r_k) = \beta \cdot PR(r_k) \quad (3.8)$$

where $PR(r_k)$ is the normalized PageRank coefficient [121], and β is a damping coefficient, which lies in the range $[0,1]$, and has been experimentally determined as equal to 0.6. Finally, the value of α in equation (3.2) has been investigated by varying between the range $[0,1]$ and the optimal value $\alpha = 0.7$ results with an optimal solution. The KB score $KB(e_j, r_k)$ is normalized in the range $[0,1]$ and is denoted as $P_{KB}(e_j, r_k)$, and is used to rank the top-k candidates.

Named Entity Linking: The final component in the Linking2Adapt pipeline is related to entity linking where based on the candidate list retrieved in the previous step, an entity mention is disambiguated with every retrieved candidate resource in the candidate resource list to decide the most suitable candidate for linking the entity mention to. To this end, based on the normalized KB scores $P_{KB}(e_j, r_k)$ obtained using equation (3.2) for every candidate resource, the entity mention is linked to the candidate with the highest (normalized) KB score following an unsupervised, greedy approach. In this way, every identified entity mention is linked with a corresponding candidate resource with the highest KB candidate score. However, mentions for which no candidate matches are retrieved from the index have been mapped to a NIL reference with an assigned type *Other* of the O_S Ontology.

Entity Mention Re-Classification: For this task, the DBpedia type (denoted by *rdf:type*) is obtained for all top-k candidate resources (which are retrieved for entity disambiguation). The DBpedia types in the DBpedia Ontology⁴ have been mapped to entity types in O_S using an ontology mapping that has been established manually. A few examples of these mappings are shown in Table 3.1. Using the DBpedia Ontology class/type for each candidate r_k retrieved in the previous step, the most suitable entity type for each entity mention is predicted by making use of the *a priori* estimation provided by the recognition component (in this case, the T-NER system) smoothed by the similarity score derived through the KB. In particular, the most probable entity type $c^*(e_j)$ for a mention is determined according to the following decision rule:

⁴<http://mappings.dbpedia.org/server/ontology/classes/>

Table 3.1: Examples of mappings between T-NER and DBpedia Ontologies

Ritter Ontology Types	Dbpedia Ontology Types
Band	Band
Company	Company, Business, Organisation, GovernmentAgency
Facility	Award, WebSite, SportFacility
Geo-Location	Place, Location, PopulatedPlace, Country, City, Locality, Region, Park
Movie	Film
Other	MeansOfTransportation, Holiday, ArtWork, Cartoon, Species, Food, Event
Person	Person, Dancer, Painter
Product	VideoGame, MusicalWork, Software, Album, Device
Sportsteam	Sportsteam, SportsClub, SportsLeague
TVshow	TelevisionShow, TelevisionEpisode, TelevisionSeries

$$c^*(e_j) = \operatorname{argmax}_c \{P_{CRF}(e_j, c(e_j)) * P_{KB}(e_j, r_k)\} \quad (3.9)$$

where $c^*(e_j)$ denotes the new entity type/class of the entity mention e_j (which is now re-classified) which may or may not be the original entity type $c(e_j)$ as estimated by the T-NER system and $P_{KB}(e_j, r_k)$ is the KB score $KB(e_j, r_k)$ normalized in the range $[0,1]$. An important point to note here is that while $P_{CRF}(e_j, c(e_j))$ score is derived by a probability distribution over a set of entity types obtained by using the CRF model, $P_{KB}(e_j, r_k)$ can be seen as a smoothing coefficient which is associated with $P_{CRF}(e_j, c(e_j))$. Thus, entity mentions are re-classified in this way based on information extracted from the Semantic Web associated with the candidate resources.

3.2.2 Entity Mention Re-Scoping

Similar to the *entity mention re-classification* approach, this is also an unsupervised approach that has been proposed with the intent of improving entity identification by *re-scoping* the boundaries of identified entity mentions based on the resources to which they are finally linked. A graphical

representation of this approach is shown in Figure 3.3(b).

Named Entity Recognition: As mentioned in the previous section, entity identification is performed by using T-NER. However, the entity types assigned by T-NER to the identified entity mentions are not considered for this approach.

Named Entity Disambiguation: A *post-processing step* has been introduced (in contrast to that in Section 3.2.1) after mentions are identified where for every identified mention which consists of capital letters (and are not present at the beginning of the mention), the mention is segmented into a set of tokens based on the capital letter. For instance, the mention ‘StarWars’ is segmented as ‘Star Wars’ so as to obtain improve candidate selection. Similar to the candidate retrieval phase in the previous section, top-k candidate resources are retrieved from the local index for every identified entity mention and a KB score is estimated for them using equation (3.2).

Named Entity Linking and Classification: Based on the KB scores $KB(e_j, r_k)$ obtained using equation (3.2) for every candidate resource, the entity mention is linked to the candidate with the highest (normalized) KB score following an unsupervised, greedy approach. In this way, every identified entity mention is linked with a corresponding candidate resource with the highest (normalized) KB candidate score achieved using equation (3.2). However, mentions for which no candidate matches are retrieved from the index have been mapped to a NIL reference with an assigned type *Thing* of another ontology used for this approach, as described below.

As mentioned above, the entity types for entity mentions as predicted by T-NER have not been taken into consideration. Moreover, a different ontology, called the *Microposts Ontology*, O_T [107], was considered for this approach so as to be able to evaluate the classification performance and compare with that in the previous section. The ontology is defined as follows:

Microposts Ontology (O_T): *Character, Event, Location, Organisation, Person, Product, Thing.*

Evidently, a manual mapping has been established between the DBpedia Ontology and the Microposts Ontology in a similar way as done between DBpedia Ontology and Ritter Ontology in the previous section. Further, the DBpedia type (i.e., the *rdf:type*) of a selected candidate resource with which the mention is linked is obtained and mapped to the corresponding entity type in the Microposts Ontology so as to perform entity classification. Every DBpedia Ontology class that could not be mapped intuitively, such as the Ontology class *Species*, has been mapped to the Microposts category *Thing*. For a specific exceptional case – DBpedia Ontology class *Name*, with its subclasses *GivenName*, *Surname*, were mapped to the Microposts category *Person* since GivenNames and Surnames are used in tweets mostly to refer to a person in the real world. This interpretation of mapping for names and surnames is inspired by work on mapping semantics in [2].

Entity Mention Re-Scoping: Based on the candidate resource selected for an entity mention to be linked to, an identified mention’s boundary is *re-scoped* according to the label of the selected candidate resource. This step is applied when the resource label is a substring of the entity mention. This step has been performed so as to reduce the identification error rate exhibited by the entity recognition system T-NER. For instance, in the tweet,

```
Day 9:Wearing a StarWars T-Shirt each day until  
‘The Force Awakens’.We’re half way there! https://t.co/QoA0xoSCJk
```

the recognition system identifies ‘StarWars T-Shirt’ as an entity mention, due to a capitalization issue, however, the linking algorithm is able to link this mention correctly with the KB resource *Star Wars*⁵, based on contextual and KB evidence. As a result, the boundary of the identified mention ‘StarWars T-Shirt’ is re-scoped to ‘StarWars’ which leads to a reduction in the error rate and an overall improvement in the identification performance of the system.

⁵http://dbpedia.org/resource/Star_Wars

3.3 Learning2Adapt

Learning2Adapt is a supervised approach, which has been proposed to adapt entity classification performed by a named entity recognizer to different ontologies (used by different named entity classifiers).

New information is continuously evolving in real-time and different ontologies and/or existing ontologies with refined levels of granularity have to be adopted so as to classify new and diverse forms of information. Further, information is being extracted today from different textual formats and from different domains ranging from scientific articles and blogposts to product reviews. For instance, articles on various **travel blogs** will have increased references to geo-locations, thus corresponding to travel ontologies, differing in terms of their levels of granularity, while information extracted from a **sports article** will adhere to a sports ontology with types such as name of the sportsteam, sportsperson/players, equipments and so on. This means that, a large number of ontologies or classification hierarchies are used today by a variety of NER systems and KB experts to identify and classify new and/or existing information. Due to the use of different ontologies for classification, however, the integration or performance evaluations of different NER systems is difficult. In such cases, a system using one (source) ontology may need to be adapted to a different (target) ontology for integrating NER systems or performance evaluations. This gives leeway to alignment or mapping among these ontologies [2, 114], which is also important from the point of view of accomplishing interoperability among them on the Semantic Web.

A recent example of integration of different NER systems is a state-of-the-art framework named NERD [106] (also described in Chapter 2), which is plugged on top of various NER systems, wherein each system is defined by its own ontology. A NERD Ontology is defined by establishing manual mappings between the entity types of the ontologies of the NER systems being integrated. Thus, in such scenarios, performing a mapping from the classification types (or entity types) in the *source domain* of one ontology to the types available in the *target domain* of another ontology is not a trivial task. The method of establishing mappings between entity types of ontologies is even more significant.

3.3.1 Ontology Mapping

As defined previously, an ontology is described as a taxonomy or a classification hierarchy with classification types or concepts based on the domain in consideration. A brief account of ontology mapping is provided in this section.

Given two ontologies A and B , **ontology mapping** is defined by obtaining a corresponding match between a concept of ontology A with the closest semantic concept of ontology B [35]. For instance, consider the ontologies in Figure 3.4. For the time being, let's assume that the Ritter Ontology, O_S [102] (as described in the previous chapter) is the source ontology with disjoint set of entity types that need to be mapped to the Microposts Ontology, O_T [104, 107] which, thus, acts as the target ontology. Thus, ontology A (i.e., the *Ritter/Source Ontology*) is defined by the concepts or the entity types *Band*, *Company*, *Facility*, *Geo-Location*, *Movie*, *Other*, *Person*, *Product*, *Sportsteam*, and *TVshow* while ontology B (i.e., the *Microposts/Target Ontology*) is defined by the concepts or entity types *Character*, *Event*, *Location*, *Organization*, *Person*, *Product*, and *Thing*. The solid blocks in Figure 3.4 depict the mappings between entity types of both the ontologies which have been established manually and is later used as a Baseline Model for the experimental evaluations, as discussed in Chapter 4. Further, the dotted blocks (shaded in blue) depict possible mappings from types in source ontology (such as the type *Person*) to types in target ontology (such as the types *Person* or *Character*), but have not been considered in the manual mappings.

The entity type, *Geo-Location* for instance, in the Ritter Ontology has been manually mapped to the closest semantic match in the Microposts Ontology which is the entity type, *Location*. An important observation to be drawn from Figure 3.4 is that, depending upon the ontologies and their respective levels of granularity, it is not always the case to have one-to-one (1:1) mappings. One-to-many (1:m) and many-to-one (m:1) mappings are possible as well. For instance, the entity types *Sportsteam*, *Company* and *Band* of the source ontology can all be mapped to the sole entity type *Organization* in the target ontology, i.e., an m:1 mapping. An example of 1:m in

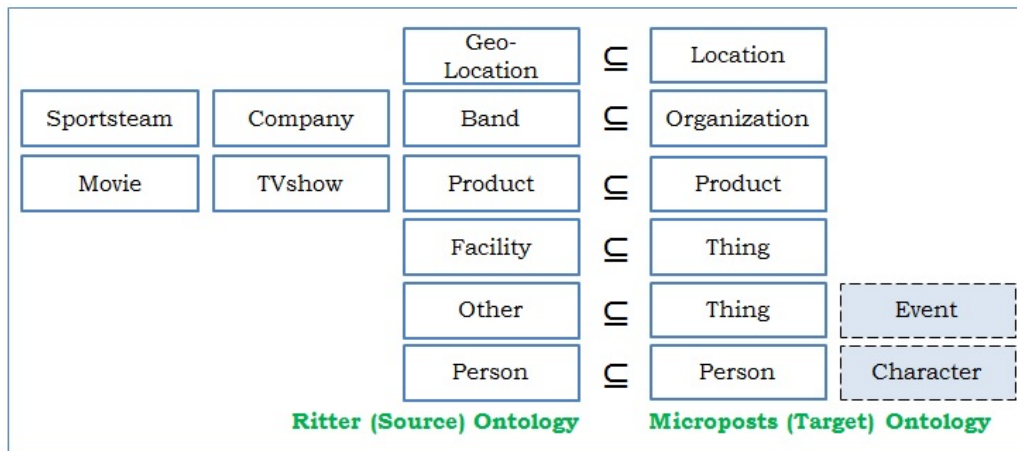


Figure 3.4: Manual Mappings between two Ontologies

this figure would be mapping of the entity type *Person* in the source ontology to entity types *Person* or *Character* in the target ontology (shown in dotted blocks in Figure 3.4). These examples in particular depict differing levels of granularity in both ontologies, where, on one hand the source ontology differentiates between entity types *Sportsteam* and *Band* which colloquially can be considered to be child concepts of the parent concept *Company*, while on the other hand, it simply uses just one higher level concept/entity type for representing human beings, i.e., *Person*. The target ontology, in contrast, uses a single higher level concept for classifying groups or companies by the entity type *Organization*, whereas it uses a parent and child concept for classifying human beings as *Person* and *Character* respectively.

Such distinctions between entity types in different ontologies depends on the application requirements, as well as the textual formats being used as input. Case in point, the textual format being considered by these ontologies here is the short, ill-formed texts of microblogs/tweets for entity classification. The source ontology is, in fact, used by a state-of-the-art NER system, T-NER [102], for entity identification and classification. One of the most ‘difficult’ to interpret (and map) entity types of this ontology is the type *Other*. While, colloquially, this type could be understood as something which cannot be classified into any other type of the source ontology, and subsequently could be mapped to the entity type *Thing* of the target ontology, on closer inspec-

tion, however, of the text phrases identified as entity mentions and classified as type *Other*, it has been found that this type could also be mapped to the type *Event* of the target ontology (shown in dotted blocks in Figure 3.4). Consider the following tweets, where entity identification and classification has been performed by T-NER [102]:

- [Xmas]*Other* cycling stocking filler ideas
@MiltonKeynesWCX @VeloPac
- RT @qikipedia: In 2011, [Saudi Arabian]*Other*
security forces detained a vulture suspected of
working for [Mossad]*Company*.

The phrases **Xmas** and **Saudi Arabian** have been identified as entity mentions and classified as type *Other*. Conventionally, the mention **Xmas** should be mapped to the entity type *Event* of the target ontology, since, it corresponds to an event in the real world and the mention **Saudi Arabian** should be mapped to the entity type *Thing* of the target ontology. However, both of these mentions have been classified as *Other* by T-NER, and thus, as per manual mappings seen in Figure 3.4, will be mapped to the entity type *Thing* of the target ontology. These examples show that use of hard-coded rules of mapping one entity type in an ontology to a specific entity type in another ontology is not always the correct solution when mapping ontologies of different NER classifiers, particularly for texts such as tweets where classification performances of NER systems are bound to suffer due to the difficulties posed by such textual formats. Thus, for this purpose, a novel supervised approach called, **LearningToAdapt** has been proposed which intends to adapt a named entity classifier of a NER system trained on a source ontology to a different ontology used by a named entity classifier in (possibly) another domain.

3.3.2 Motivation

In order to identify and classify entity mentions, most of the state-of-the-art models [74, 102, 134] leverage different kinds of contextual information available in the analysed text (e.g., the use of articles and/or prepositions before the entity mention), as well as in other corpora (e.g., the distribution of words across different classes of entity mentions represented in a knowledge base). While contextual information that can be used by Natural Language Processing methods is quite reliable and abundant in well-formed texts (yielding to good accuracy of entity classification methods), it is less reliable and scarcer in microblogging environments [31]. As a result, entity classification is often inaccurate for texts obtained from microblogging platforms.

In addition to the intrinsic difficulty of entity classification, it is often the case that different NER systems use different ontologies for entity classification, and a system using one source ontology may need to be adapted to use a different target ontology according to application requirements. To the best of our knowledge, state-of-the-art approaches have mostly performed the task of mapping concepts/entity types of ontologies manually [106], or have used similarity measures to find concepts from a source ontology that could be mapped to corresponding concepts in the target ontology [34, 117]. A first investigation aimed at dealing with this issue has been presented in [106], as indicated previously, where manual mappings between ontologies has been defined. Although this study represents a fundamental step towards the definition of cross-domain NER systems, some open problems need to be accurately addressed:

1. **Mention Mis-classification:** Entity mentions are often mis-classified (or incorrectly classified) by a NER system due to two main reasons:
 - not enough instances available in the training set,
 - the training set is characterized by an unbalanced distribution over the ontology types.

Consider, for instance, an entity mention “Olaf” (a movie character) which has been erroneously classified by T-NER as *Geo-Location* in the

source domain instead of *Person*, and should be mapped to *Character* in the *target domain*. Given a deterministic manual mapping, as the one reported in Figure 3.4, the mention would eventually be mapped to the type *Location* in the target domain.

2. **Type Uncertainty:** There are also cases where a text phrase referring to an entity mention may be polysemous (as exemplified in Section 2.3.1), which further complicates the mapping decisions. While well-structured texts provide meaningful insights into the contextual usage of a mention, there can still be cases where it is difficult for an entity recognition system to classify a mention correctly. Consider, for instance, the polysemous text phrase “Starbucks” in a well-structured document snippet:

Millie Bobby Brown, best known for her role as Eleven in the Netflix series *Stranger Things*, showed off her vocal talent in a silly clip for Starbucks in the drive-thru...

"Can I have a venti latte / And a caramel frappuccino? / Oh, please!" she can be heard singing below, swapping her words in place of the original lyric, "Hello from the other side / I must have called a thousand times / To tell you...", and then...

Here, the mention can either be classified as a *Geo-Location* (a particular Starbucks shop), or a *Product* (the Starbucks coffee). On the other hand, the matter of correctly classifying a polysemous text phrase in short textual formats (such as microblog posts) tends to become challenging. Due to the concise nature of a microblog post, it is difficult for a NER system to decide the entity type (in source ontology) for a mention whose entity type may range from, for instance, *Geo-Location*, *Product* to *Company*. This is exemplified in Figure 3.5. The mapping decision becomes difficult for an expert as well, since *Geo-Location* would be mapped to *Location*, and *Company* would be mapped to *Or-*

ganization, according to the manual mappings (see Figure 3.4).

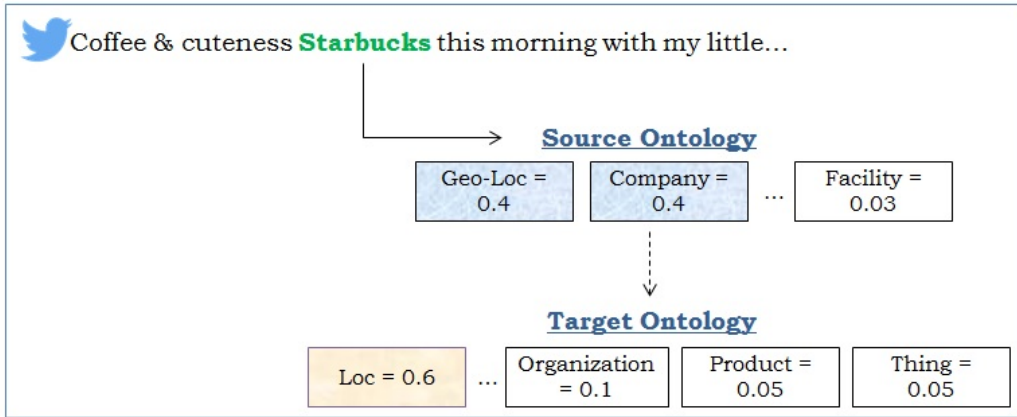


Figure 3.5: Example: Ambiguous Entity Mentions

3. **Fork Mappings:** There are cases where mentions classified as one type in the source ontology should be cast into one of two different types of the target ontology. In this work, such mappings are referred to as **Fork Mappings**. For the purpose of experimental investigations, two such cases of fork mappings have been identified between Ontology A (Ritter Ontology) and Ontology B (Microposts Ontology) as indicated by the dotted blocks in Figure 3.4:

- when a mention of type *Person* in the source ontology should be mapped to *Person* or *Character* in the target ontology,
- when a mention of type *Other* in the source ontology should be mapped to *Thing* or *Event* in the target ontology,

In this case, in order to re-classify entity mentions based on a deterministic manual mappings model, an expert has to select one target type for each source type involved in a fork mapping, which can be an error-prone and a subjective decision.

Thus, in order to tackle the above mentioned issues the approach **Learning-ToAdapt** has been proposed. A noteworthy point is that both the ontologies

that have been considered here (Figure 3.4) are used in the state-of-the-art for classifying entity mentions recognized from microblog posts [102, 104]. Although, this approach for mapping entity types from a source to a target domain has been experimented for microblog posts, it can be applied to other textual formats as well. The main motivation underlying the proposed approach is concerned with two main issues:

1. The amount of data available in the target domain may be limited to accurately train a NER system. A major assumption in many NER systems based on machine learning (e.g. Conditional Random Fields [69] and Labelled LDA [96]) is that the training and test data must be in the same feature space and have the same distribution. However, when a NER system needs to be adapted to a new domain ontology, this assumption may not hold. For example, one may need to have a NER system in a target domain of interest, but might only have sufficient training data in a source domain of interest, where the latter data may be in a different feature space or follow a different data distribution.
2. The training of a NER system based on a new complex target ontology can be expensive in terms of time and labor. Many NER systems are grounded on complex ontologies to train a probabilistic model able to recognize and classify entity mentions in a given domain (e.g. the system (T-NER) proposed in [102] trains LabelledLDA using Freebase as underlying domain ontology, which is composed of more than 39 million real-world entities). Moreover, the dimension of knowledge bases increases rapidly thanks to the new upcoming entities evolving every day. In this case, it could be very expensive to re-train any NER model on either updated or brand new ontologies.

3.3.3 Problem Formulation

The problem of mapping the types of entity mentions from a source domain to the types in a target domain can be viewed as a machine learning problem. In particular, given a set of entity mentions identified by a Named Entity Recognition model originally trained in a source domain, the main goal is to learn how to map the source type probability distribution to the target one.

More precisely, let R_S be a Named Entity Recognition (NER) model trained on a set $\Omega_S = \{s_1, s_2, \dots, s_n\}$ of entity mentions annotated according to a source ontology O_S . Let $\Omega_T = \{e_1, e_2, \dots, e_m\}$ be a set of entity mentions that needs to be automatically labeled according to a target ontology O_T , by using the NER model R_S previously trained on Ω_S .

The labeling of Ω_T using R_S can be viewed as a transfer learning problem [92]. In particular, the main goal is to learn a target predictive function $f(\cdot)$ in Ω_T using some knowledge both in the source domain S and the target domain T . More formally, let $P(\Omega_T, O_S)$ be the distribution in the source domain used to label an entity mention $e_j \in \Omega_T$ with the most probable type $c_S^* \in O_S$ according to R_S . Let $P(\Omega_T, O_T)$ be the distribution in the target domain that needs to be discovered given the knowledge about $P(\Omega_T, O_S)$ and the target type $c_T \in O_T$.

$P(\Omega_i, O_S)$ from R_S $g(P(\Omega_i, O_S)) = c_T \in O_T$

Entity Mention	Source Type	P(Facility)	P(Person)	P(Organization)	P(Band)	P(Movie)	P(TVShow)	Target Type
Paris B.O.	Facility	0.36	0.34	0.05	0.1	0.05	0	Location
Harry Potter	Person	0	0.5	0	0.2	0.1	0	Character
@EF	Organization	0	0.1	0.7	0.2	0	0	Organization
...	

P(Location)	P(Character)	P(Person)	P(Organization)	P(Event)	P(Product)
0.8	0.05	0.05	0.1	0	0
0	0.6	0.2	0.1	0	0.1
0	0.4	0.5	0.1	0	0
...

$P(\Omega_i, O_T)$

Figure 3.6: Example of Input-Output Space in Learning2Adapt

Now, the labeling of Ω_T using R_S can be modeled as a learning problem aimed at seeking a function $\phi : P(\Omega_T, O_S) \rightarrow P(\Omega_T, O_T)$ over the hypothesis space Φ . In this case, it is convenient to represent ϕ as a function $f : P(\Omega_T, O_S) \times P(\Omega_T, O_T) \rightarrow \mathbb{R}$ such that:

$$g(P(e_j, c_S)) = \arg \max_{c_T \in O_T} f\left(P(e_j, c_S), P(e_j, c_T)\right) \quad (3.10)$$

In order to address this problem, an input space needs to be created which represents each entity mention e_j that can be used for learning to map the predicted source type $c_S \in O_S$ to the target type $c_T \in O_T$. As mentioned before, the input space corresponds to $P(\Omega_T, O_S)$, and in particular to the explicit distribution given by R_S for each entity mention. The output space denotes the most probable type $c_t \in O_T$, and more specifically, the type distribution $P(\Omega_T, O_T)$ in the target domain. A simple example of the input and output space for Learning2Adapt is reported in Figure 3.6.

Now the goal is to determine the function f that is able to correctly label an entity mention $e_j \in \Omega_T$ according to the prediction $P(e_j, c_S)$ given by a NER model previously trained on Ω_S . To accomplish this task, several learning algorithms have been considered:

- **Naïve Bayes** (NB) [77] is the simplest generative model that can be used for Learning2Adapt. It predicts the target type c_T of a given entity mention e_j given a vector representation of the distribution $P(e_j, O_S)$ by exploiting the Bayes' Theorem:

$$\begin{aligned} P(e_j = c_T \mid P(e_j, c_{S_1}), \dots, P(e_j, c_{S_n})) &= \\ &= \frac{P(c_T)P(P(e_j, c_{S_1}), \dots, P(e_j, \dots, c_{S_n}) \mid e_j = c_T)}{P(P(e_j, c_{S_1}), P(e_j, c_{S_2}), P(e_j, \dots, c_{S_n}))} \end{aligned} \quad (3.11)$$

Thus, the final target type c_T^* is determined according to the following maximum a posteriori (MAP) decision rule:

$$c_T^* = \arg \max_{c_T} P(e_j = c_T) \prod_{k=1}^{|O_S|} P(c_T)P(P(e_j, c_{S_k}) \mid t_i = c_T) \quad (3.12)$$

- **Support Vector Machines** (SVM) [24] are linear learning machines aimed at determining the optimal hyperplane that discriminate samples of different types. When samples are not linearly separable, that is when a hyperplane able to separate them does not exist, features are transformed into a higher dimensional space that allow the “new samples” to be better separated. Consider a non-linearly separable training set D defined over the input space $I = P(\Omega_T, O_S)$ and with an output

class domain O_T , as defined by

$$D = \left(\left(P(e_1, c_{S_1}), c_{T_1} \right), \dots, \left(P(e_m, c_{S_n}), c_{T_r} \right) \right) \quad (3.13)$$

the main goal is to map non-linear training data $(P(e_j, c_{S_1}), c_{T_k})$ from \mathbb{R}^n into a new feature space \mathbb{F} by a kernel function. Support Vector Machines then find the optimal hyperplane:

$$H = \{(P(e_j, c_S), c_{T_k}) \in \mathbb{R}^n : w \cdot ((P(e_j, c_S), c_{T_k})) + b = 0\} \quad (3.14)$$

with the maximum margin, i.e. with the maximum distance between class samples. The optimal hyperplane H is defined by learning from data two parameters: the weight vector w and the bias b .

- **Decision trees** (DT) [16] are classifiers presented as binary tree-like structure, where each node corresponds to a variable in the input space and edges represent possible realization of that variable. Since this classifier outputs a dichotomic decision tree, it can be used to determine the type of unclassified entity mentions by considering its descriptive attribute realizations, i.e $P(\Omega_T, O_S)$.

Building a decision tree model from a training dataset involves two phases. In the first phase, a splitting attribute and a split index are chosen. The second phase involves splitting the records among the child nodes based on the decision made in the first phase. This process is recursively continued until a stopping criterion is satisfied. The choice about the variable ordering (from the root to the leaf) and the values for the splitting rule is a critical aspect. The most widely used indices, for evaluating whether a node should be split or not is Gini Index. Given a variable $j \in P(\Omega_T, O_S)$ with t hypothetical realizations, the Gini Index I_G is defined as

$$I_G(j) = 1 - \sum_{t=1}^m \alpha(j, t)^2 \quad (3.15)$$

where $\alpha(j, t)$ represents the frequency of the t value in the j variable.

- ***K-Nearest Neighbor*** (KNN) [33] assigns a sample $P(e_j, c_S)$ to a target type c_T that is the most common one amongst the most similar K samples into the training data. This classifier is based on one of the most well known distance metric, i.e. the Euclidean Distance, to identify the most similar instances. Given a vector $P(e_j, c_S)$ of an entity mention that needs to be mapped and a training instance vector $P(e_j, c_S)$, their Euclidean Distance is computed as follows:

$$\begin{aligned} dist(P(e_j, c_S), P(e_j, c_S)) &= \\ &= \sqrt{\sum_{k=1}^{|O_S|} \left(P(e_j, c_{S_k}) - (P(e_j, c_{S_k})) \right)^2} \end{aligned} \quad (3.16)$$

The final target type for an entity mention e_j is selected through a simple voting among the K most similar training examples.

- ***Bayesian Networks*** (BN) [93] are probabilistic graphical models that compactly represent the joint probability distribution of a set of random variables. The main assumption, captured graphically by a dependency structure, is that each variable is directly influenced by only few others. A probability distribution is represented as a directed acyclic graph, whose nodes represent random variables and whose edges denote direct dependencies between a node $j \in (P(\Omega_T, O_S), O_T)$ and its set of parents $Pa(j)$. Formally, a Bayesian Network asserts that each node is conditional independent of its non-descendants given its parents. This conditionally independence assumption allows one to represent concisely the joint probability distribution:

$$\begin{aligned} P\left(\left(P(e_1, c_{S_1}), c_{T_1}\right), \dots, \left(P(e_m, c_{S_n}), c_{T_r}\right)\right) &= \\ &= \prod_{i=1} P\left(\left(P(e_j, c_S), c_T\right) \mid Pa\left(\left(P(e_j, c_S), c_T\right)\right)\right) \end{aligned} \quad (3.17)$$

where $P(\cdot \mid Pa(\cdot))$ is described by a conditional probability distribution (CPD).

- **Multi Layer Perceptron** (MLP) [109] is one of the most widely used models in the general class of artificial neural networks. A MLP provides a non-linear mapping from a real-valued input vector $P(e_j, c_S)$ to an output $c_T \in O_T$. The basic idea is that a vector of input values is multiplied by a weight matrix W , and the resulting values are each individually transformed by a non-linear function to produce “hidden node” outputs. The produced output is then transformed again following the same procedure, leading to a process that continues until a given number of layers is obtained. Given a set D of training examples (see Equation (3.13)), for each single layer b_j MLP learns the function

$$b_j = \sum_i^{|\Omega_T|} w_i \cdot P(P(e_j, c_S)) \quad (3.18)$$

where w_i are the model parameters that need to be learned.

- **Multinomial Logistic Regression** (MLR) [70] is a classification method that generalizes logistic regression to multi class problems. The principle underlying a MLR is to construct a linear predictor function that allows to predict the target type starting from a set of weights that are linearly combined with the explanatory variables (features):

$$score(P(\Omega_T, O_S), O_T) = \beta \cdot P(\Omega_T, O_S) \quad (3.19)$$

where β are parameters to be learned. In the multi class case, the training algorithm uses the one-vs-rest scheme.

3.3.4 Contextual Evidence for Learning2Adapt

As mentioned previously, various state-of-the-art systems use different kinds of contextual information available from the text and/or the KB under consideration for entity recognition and entity linking. Therefore, the input space for entity mentions identified from a set of tweets is created by two different methods in Learning2Adapt:

- **Use of contextual information** surrounding an entity mention in a tweet (hereafter, referred to as *With Context*, or *W.C. setting*): In this

method, the input space used for LearningToAdapt is created using the distribution $P(\Omega_T, O_S)$ (in the source domain) for an entity mention $e_j \in \Omega_T$ with the most probable type $c_S^* \in O_S$ according to R_S as follows:

$$P(\Omega_T, O_S) = \frac{\sum_{i=1}^n (P(w_i, c_m)) + P(e_j, c_m)}{n + 1} \forall c_m \in O_S \quad (3.20)$$

where $P(w_i, c_m)$ denotes the probability distribution of a word w_i surrounding the entity mention e_j and c_m denotes an entity type in the source ontology O_S . The maximum number of words that are considered to be in the *context window* is 6, i.e., $i = \{1, \dots, 6\}$ in a way that equal number of words from both sides of a mention, left and right, are considered so as to better investigate the influence of words within the context window on the mention’s type. Here, the term *context window* refers to a text window which is supposedly used to deduce a mention’s type based on its contextual usage in the tweet. For instance, consider the tweet:

<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <p>Can't wait for the next Harry Potter.</p> <p style="color: blue; font-size: small;">Left Window</p> </div> <div style="text-align: center;"> <p>Amazing movie!!!</p> <p style="color: blue; font-size: small;">Right Window</p> </div> </div>
--

The mention **Harry Potter** is a polysemous term, in the sense that it could be of the entity type *Movie*, *Person* (which denotes the fictional character of the movie) or *Product* (which denotes the book series) according to the Source Ontology. On the other hand, if words in the context window surrounding the mention, i.e., “for the next **Harry Potter**. *Amazing movie*” are considered, then it can be deduced with a higher certainty that the mention is of type *Movie*. However, in many cases, words within the context window maybe not be separated by a white space (due to presence of punctuation marks) such as:

<div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <p>top-of-the-morning here in Paris!</p> <p style="color: blue; font-size: small;">Left Window</p> </div> <div style="text-align: center;"> <p>Lovely city, Mr.Richards.</p> <p style="color: blue; font-size: small;">Right Window</p> </div> </div>

Although a maximum of three words are considered from each side of the mention, in this case the probability distributions

for $\{top-of-the-morning, here, in\}$ from the left context window, the probability distribution for the mention itself and for the words $\{Lovely, city, Mr.Richards\}$ from the right context window will be considered. The overall distribution $P(\Omega_T, O_S)$ is, further, determined by considering every word’s probability distribution in the context window along with the distribution of the mention itself, in a way that a mean of probabilities of all the words and the mention for every entity type $c_S^* \in O_S$ according to R_S is calculated. Further, the most probable source type $c_S^* \in O_S$ for the mention e_j is derived as follows:

$$c_S^* = \arg \max_{c_m} P(\Omega_T, O_S) \quad (3.21)$$

- **Use of no contextual information** from the surrounding of an entity mention in a tweet (hereafter, referred to as *Without Context*, or *Wo.C. setting*): In this method, the input space is created by using a probability distribution of an entity mention $e_j \in \Omega_T$ independent on the probability distribution of words in the context window. The probability distribution for Wo.C. method is denoted by $P'(\Omega_T, O_S)$ and is used to derive the most probable source type $c_S^* \in O_S$ as follows:

$$c_S^* = \arg \max_{c_m} P'(\Omega_T, O_S) \quad (3.22)$$

3.4 Learning2Link

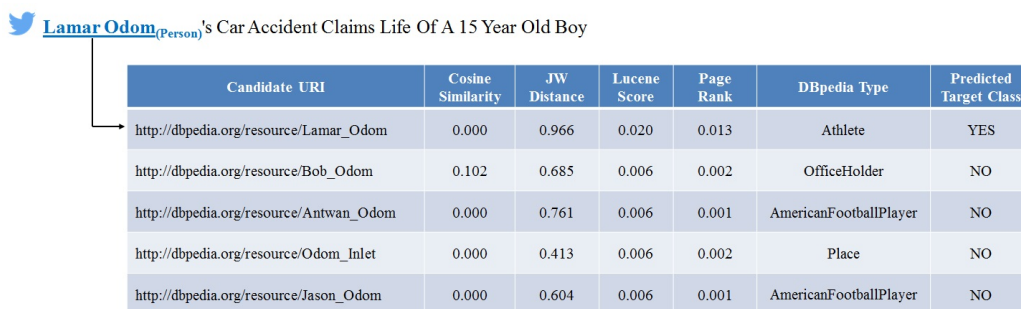
Learning2Link is a supervised approach, which has been proposed to adapt entity linking by using evidence provided by entity recognition and disambiguation. A **Decision Criteria** has been used in this approach in order to decide if an entity mention is linkable or unlinkable. If the mention is linkable, then it is linked to the most suitable candidate resource retrieved from the KB using the supervised linking approach discussed below.

Named Entity Recognition: In order to identify entity mentions from a set of tweets, T-NER has been again used, as also in the previous

approaches. Based on this model, every identified entity mention e_j can belong to one or more entity types $c(e_j)$ of ontology O_S , each associated with a probability denoted as $P_{CRF}(e_j, c(e_j))$ in equation (3.1), which illustrates the probability that an entity mention e_j belongs to an entity type $c(e_j)$. The output of CRF is a set of entity mentions e_1, e_2, \dots, e_m identified from a given tweet t . In order to distinguish the entity types predicted for an entity mention in the further phases, the entity type predicted in this step for an entity mention e_j is referred to as the $T - NER_{type}$.

Named Entity Disambiguation and Linking: A supervised learning approach has been used for entity linking in tweets, as opposed to the unsupervised one as described previously in Section 3.2. Several factors such as lexical similarity of an entity mention with a candidate resource, cosine similarity, and popularity of a candidate resource are taken into consideration for creating an input space for an entity mention so as to be able to link it with an appropriate candidate resource from the list of top-k resources retrieved from the KB.

Starting from the coefficients estimated in equation (3.2), which denote how similar is an entity mention e_j to a given candidate resource r_k , an input space is derived that is able to allow any machine learning model to learn if the mention e_j is linkable or not linkable to r_k . An example of the derived input space is represented in Figure 3.7.



The figure shows a tweet from Lamar Odom: "Lamar Odom's Car Accident Claims Life Of A 15 Year Old Boy". Below the tweet is a table with 7 columns: Candidate URI, Cosine Similarity, JW Distance, Lucene Score, Page Rank, DBpedia Type, and Predicted Target Class. The table lists five candidate resources from DBpedia, with the first one (Lamar Odom) being the predicted target class.

Candidate URI	Cosine Similarity	JW Distance	Lucene Score	Page Rank	DBpedia Type	Predicted Target Class
http://dbpedia.org/resource/Lamar_Odom	0.000	0.966	0.020	0.013	Athlete	YES
http://dbpedia.org/resource/Bob_Odom	0.102	0.685	0.006	0.002	OfficeHolder	NO
http://dbpedia.org/resource/Antwan_Odom	0.000	0.761	0.006	0.001	AmericanFootballPlayer	NO
http://dbpedia.org/resource/Odom_Inlet	0.000	0.413	0.006	0.002	Place	NO
http://dbpedia.org/resource/Jason_Odom	0.000	0.604	0.006	0.001	AmericanFootballPlayer	NO

Figure 3.7: Example of Input Space in Learning2Link

This input space is used for training various learning algorithms such as *Decision Trees (DT)* [16], *Multi-Layer Perceptron (MLP)* [109], *Support*

Vector Machines (SVM) with Linear, Polynomial and Radial kernels [24], *Bayesian Networks (BN)* [93], *Voted Perceptron (VP)* [45], *Logistic Regression (LR)* [77] and *Naïve Bayes (NB)* [77]. The target class is a boolean variable which indicates whether or not a candidate resource r_k is suitable for an entity mention e_j to be linked to.

Algorithm 1 : Decision Criteria

if entity mention e_j is linkable to a single unique candidate resource r_k with predicted target class *Yes* **then**
 Link the mention e_j to the resource r_k
else if entity mention e_j is linkable to more than one candidate resources with predicted target class *Yes* **then**
 Link the mention e_j to candidate resource r_k with the highest probability
else if all candidate resources for entity mention e_j have predicted target class *No* **then**
 Set the mention e_j as a NIL mention
end if

A **Decision Criteria** is then used to determine if a mention is *linkable* or *unlinkable* based on the predicted target classes. If a mention is linkable, the criteria is used to predict the most suitable candidate resource from a list of candidate resources $\{r_1, r_2, \dots, r_k\}$ for an entity mention e_j (or detect the NIL mentions, if the mention is unlinkable). The decision criteria is described below in Algorithm 1, according to which a mention is *linkable* iff one or more candidate resources for the mention have a predicted target class *Yes* and is *unlinkable* iff all the candidate resources for the mention have a predicted target class *No*. If a linkable mention e_j has more than one predicted target class *Yes*, then it is linked to the candidate resource r_k with the highest probability, as determined by the learning model. Finally, the entity type of a mention is determined by the DBpedia type of the selected candidate resource, which is further mapped to a type in Ontology O_T based on an Ontology mapping that has been developed between the O_T and the DBpedia Ontology manually. In case a mention is *unlinkable*, therefore, corresponding to a NIL mention, the entity type is determined according to

the $T - NER_{Type}$ derived in the entity recognition phase.

The experimental results for the NEEL framework and the approaches discussed in this chapter have been presented in the next chapter.

4 EXPERIMENTAL ANALYSIS

This chapter presents the experimental analysis of the approaches described in Chapter 3 in sections 4.1, 4.2 and 4.3 for Linking2Adapt, Learning2Adapt and Learning2Link respectively.

4.1 Experimental Analysis: Linking2Adapt

This section presents an experimental analysis of the **Linking2Adapt** approach that has been proposed in Chapter 3 Section 3.2 with the intent of improving the performance of entity recognition by the use of entity linking, thereby, improving the overall performance of the proposed NEEL framework.

4.1.1 Entity Mention Re-Classification

This section presents an experimental analysis of the *entity mention re-classification* approach (discussed under Linking2Adapt in Section 3.2.1), which has been proposed in order to improve entity classification performed by the recognition component by means of a feedback method so as to *re-classify* the identified entity mentions based on evidence obtained from the linking component in the proposed NEEL framework. A schematic view of this approach is shown in Figure 4.1.

Experimental Setup: A gold-standard corpus of tweets (in English language) made available by Ritter et al. [102] has been used for the experiments for recognition and linking phases of the framework. This dataset consists of ≈ 2400 tweets with 47k+ tokens. Additionally, for performing a detailed analysis, a manually curated set of 1616 entity mentions (referred to as the **Ground Truth**) has been prepared from the given corpus of tweets, and annotated with the entity types (as shown below). This ground truth has been populated with those entity mentions whose real world

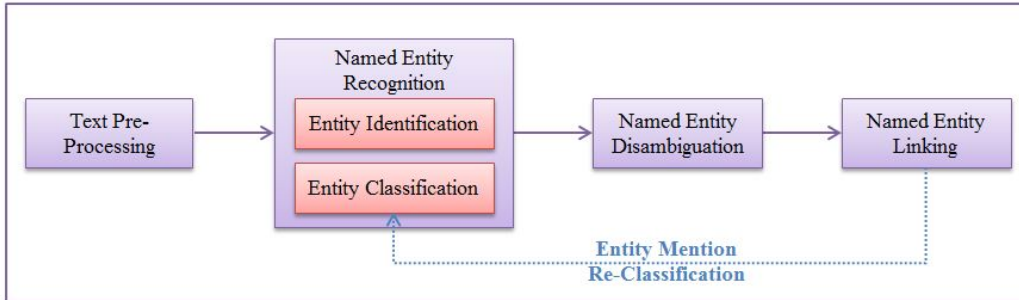


Figure 4.1: Entity Mention Re-classification

Table 4.1: Type-Wise Distribution (%): Ground Truth vs T-NER

Entity Type	Ground Truth Distribution	T-NER Distribution
Band	4.52	7.15
Company	10.64	9.16
Facility	7.18	4.88
Geo-Location	17.88	16.24
Movie	2.66	8.02
Other	15.53	10.83
Person	28.65	34.63
Product	7.18	4.55
Sportsteam	3.59	1.94
TVshow	2.17	2.61

counterparts exist in the Unstructured Web. The type-wise distribution of entity mentions in the ground truth is reported in Table 4.1.

Named Entity Recognition: For the task of identifying and classifying entity mentions, a state-of-the-art CRF-based system, called T-NER [102] has been used, which is defined by its ontology, i.e., the *Ritter Ontology* (denoted by O_S), as stated in the previous chapter. T-NER segments tweets in the gold standard into text phrases where each text phrase is denoted as an entity mention or a non-mention. Theoretically, an entity mention is a text phrase which potentially denotes a named entity in the real world.

Table 4.2: T-NER: Identification Performance Analysis

Text Phrase	Identification Analysis (w.r.t Ground Truth)		Example Mentions	Accuracy (%)
Entity Mentions (1496)	Identified Entity Mentions	Correctly Identified	Anna Wintour	91.46
		Incorrectly Identified	Indian	1.11
	Missed Entity Mentions		NFL	7.43

In terms of text phrases identified as entity mentions by T-NER, an *entity mention* refers to a text phrase in a tweet which denotes a named entity in the real world and is thereby associated with an entity type and a probability given by $P_{CRF}(e_j, c(e_j))$ as stated in equation (3.1). On the other hand, theoretically, a non-mention refers to a text phrase that does not denote any relevant named entity in the real world. Additionally, an entity mention can be:

- a *correctly identified mention* if it exists in the ground truth and refers to a named entity in the real world and has been identified as a mention by the NER system, such as the journalist *Anna Wintour* (in Table 4.2); or
- an *incorrectly identified mention* if does not exist in the ground truth and/or does not refer to a named entity in the real world but has been identified as a mention by the NER system, such as the text phrase *Indian* (in Table 4.2); or
- a *missed entity mention* if it exists in the ground truth and refers to a named entity in the real world but the named entity recognizer has failed (or missed) to identify it as a mention, and instead identified it as a non-mention, such as the mention *NFL* (in Table 4.2).

T-NER identifies a total of 1496 text phrases as entity mentions, while the remaining 44k text phrases have been identified as non-mentions (in contrast to the 1616 mentions present in the ground truth). Table 4.2 presents an identification performance analysis of T-NER, where out of 1496 mentions identified by T-NER, 91.46% are correctly identified while 1.11% are incorrectly identified w.r.t the mentions in the ground truth. Further, it fails to identify 7.43% of entity mentions, as observed by the missed entity mentions

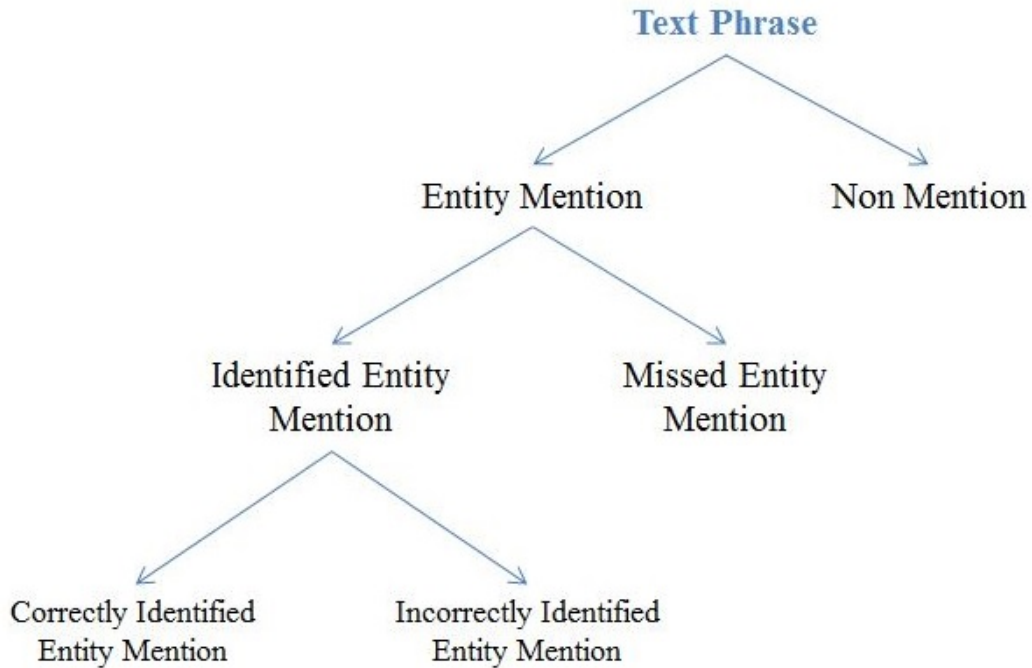


Figure 4.2: Identification Scenarios

in the table. This can be explained by the fact that a large number of new entities emerge constantly on the Web as well as on social media, before a knowledge base can index them. This causes a supervised entity recognition system (such as T-NER) to fail to identify such named entities. For the sake of simplicity, a graphical representation of all identification scenarios is shown in Figure 4.2.

Further, the text phrases identified as entity mentions are classified into entity types as per the Ritter Ontology, as stated in the previous chapter. As stated previously, Table 4.1 shows a type-wise distribution of (1616) entity mentions in the ground truth while the distribution of (1496) mentions identified and classified by T-NER has also been reported in this table. It can be observed that while for some entity types (such as *Company*, *Facility*, *Other* and so on), T-NER under-classifies the entity mentions into the respective types as compared to those in the ground truth, it over-classifies entity mentions for the entity types *Band*, *Movie*, *Person*, and *TVshow* on the other hand. This can be attributed to the entity classification and segmentation

Table 4.3: T-NER: Classification Performance Analysis

Text Phrase	Classification Analysis	Example		Classification Accuracy (%)
		Entity Mention	Entity Type	
Entity Mentions (1496)	Correctly Classified	Justin Bieber	Person	61.56
	Incorrectly Classified	Chicago	Person	37.97
	Segmentation Error	Alpha, Omega	Geo-Location, Band	0.47
Non-Mentions (44,792)	Correctly Classified	It	O	99.80
	Incorrectly Classified	justthen	Person	0.20

errors as can be observed in Table 4.3. This table presents a detailed analysis of classification performance of T-NER (in terms of accuracy). As indicated in the table, a text phrase can be:

- a *correctly classified entity mention* if it is *correctly identified* (as described above) and T-NER correctly classifies it into an entity type (as per the ground truth), such as correctly identified mention *Justin Bieber* which has been classified into the type *Person* as observed in Table 4.3; or
- an *incorrectly classified entity mention* if it is *correctly identified* by T-NER but classified incorrectly (as per the ground truth), such as the correctly identified mention *Chicago* being classified incorrectly as the type *Person* instead of the type *Geo-Location* as observed in Table 4.3.

Out of the text phrases identified as entity mentions, 61.56% are *correctly classified* while 37.97% of them are *incorrectly classified*, as shown in Table 4.3. High identification accuracy (in terms of *correctly identified mentions* in Table 4.2) and classification accuracy (in terms of *correctly classified mentions* in Table 4.3) can be attributed to the fact that T-NER is a supervised system which has been trained on the given gold standard of tweets. This performance is subject to depreciation if other Twitter datasets are used (as will be seen in the succeeding sections).

Text phrases that are ideally entity mentions (i.e., as per the ground truth) also suffer segmentation issues, in that, T-NER exhibits segmentation errors as a result of which text phrases are not correctly segmented (due

to which the entity mentions are not correctly identified and/or classified). Such text phrases which are ideally entity mentions and suffer with segmentation problems are indicated in Table 4.3 as *segmentation errors*; such as the (single) mention *Aplha-Omega* which, due to a segmentation error, has been identified as two distinct mentions, *Alpha* and *Omega*, and thus incorrectly classified as *Geo-Location* and *Band* respectively, instead of being classified as *Movie*. As shown in Table 4.3, a small percentage of entity mentions (0.47%) suffer *segmentation errors*. Further, *missed entity mentions* (as explained above) are classified as *O* where *O* is used to denote a non-mention as per the IOB encoding. On the other hand, a non-mention can be:

- a *correctly classified non-mention* if it does not refer to any named entity in the real world and has not been identified as a mention by the NER system (such as a text phrase *It* being classified as *O*); or
- an *incorrectly classified non-mention* if it does not refer to any named entity in the real world but has been identified and classified as a mention by the NER system (such as text phrase *justten* which has been classified as the type *Person*, instead of *O*).

As evident from Table 4.3, most of the non-mentions are correctly classified as such (99.80%) whereas a very small percentage (0.20%) is incorrectly classified. This can be attributed to the use of unnecessary capital letters, misleading presence or absence of punctuation marks or tokenization errors which can very easily confuse a supervised entity recognition system such as T-NER. For the sake of simplicity, a graphical representation of all the afore-mentioned classification scenarios is shown in Figure 4.3. Finally, type-wise performance analysis of T-NER for the gold standard corpus of tweets is shown in Table 4.5 (under the column **T-NER Performance Analysis**) where:

$$Precision(P) = \frac{|\{cor.cl\} \cap \{cl\}|}{|\{cl\}|} \quad (4.1)$$

$$Recall(R) = \frac{|\{cor.cl\} \cap \{cl\}|}{|\{cor.cl\}|} \quad (4.2)$$

$$F - Measure(F_1) = \frac{2 \times P \times R}{P + R} \quad (4.3)$$

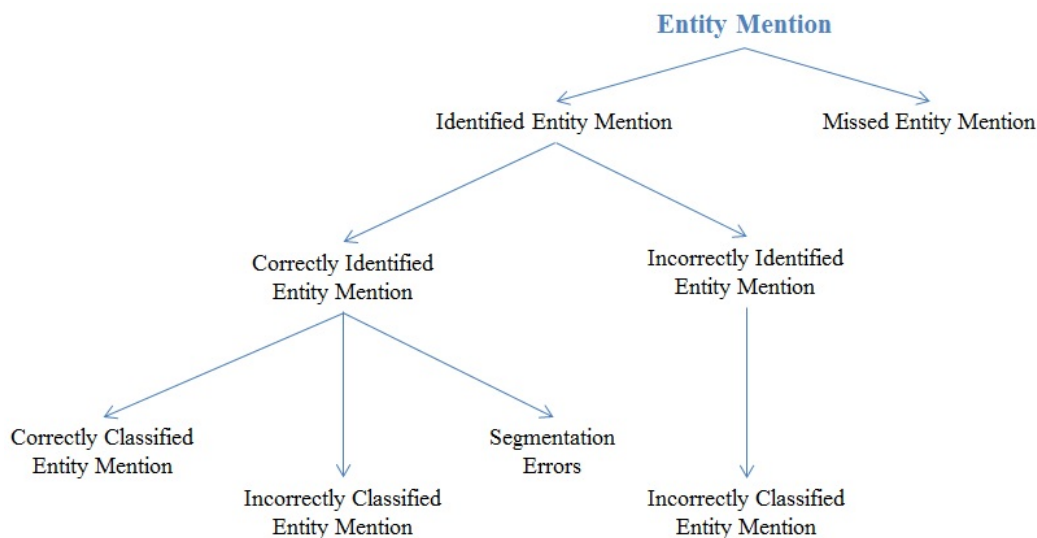


Figure 4.3: Classification Scenarios

Here *cor.cl* denotes correctly classified entity mentions, while *cl* denotes mentions that have been identified and classified into one of the T-NER entity types. It can be observed that T-NER exhibits low precision for the entity types *Band* and *Movie* in particular, which can be attributed to smaller shares of these types in the overall type distribution for the mentions identified by T-NER, as seen in Table 4.1.

Named Entity Disambiguation: A local index has been prepared for this step, for which the DBpedia dataset of ‘Lables’ (which comprises of ≈ 4.5 M things¹) has been indexed, which serves as a look-up repository for retrieving (top-k) candidate matches for entity mentions that have been identified in the previous step. Disambiguation and Page Re-Direct information for the afore-mentioned Lables has also been incorporated in the local index. ≈ 4 k candidate KB resources are retrieved (using $k = 3$) for a total of 1442 entity mentions (out of 1496). These candidates are ranked based on their corresponding knowledge base scores $KB(e_j, r_k)$ computed using equation (3.2). No candidate resource could be retrieved for the remaining mentions. The detailed analysis of this step is summarized in Table 4.4.

¹<http://wiki.dbpedia.org/dbpedia-data-set-2015-04>

Table 4.4: Entity Linking Performance Analysis

Entity Mention	Linking Analysis	Example		Linking Accuracy (%)
		Entity Mention	DBpedia Type	
Linkable	Correctly Linked	Wisconsin	Geo-Location	63.12
	Incorrectly Linked	America	Movie	3.05
	Mis-Linked	N.J.	Thing	16.15
Unlinkable	Mis-Linked	Secrets	Thing	14.07
	OOKB	Widro	Thing	3.61

An identified entity mention can be:

- a *linkable mention* if it refers to a text phrase that has been correctly identified as a mention by the NER system (in a NEEL framework) and can be linked with an appropriate existing resource in the KB (which eventually refers to a named entity in the real world); or
- an *unlinkable mention* if it refers to a text phrase that has been correctly identified as a mention by the NER system (in a NEEL framework) but cannot be linked to any resource in the KB, due to several factors which are discussed below.

Further, a linkable mention can be:

- a *correctly linked mention* if it has been correctly identified, and correctly disambiguated with the most suitable KB resource which represents the correct named entity for the given mention in the real world, such as the correctly identified mention *Wisconsin* that has been disambiguated with the correct KB resource <http://dbpedia.org/resource/Wisconsin> of DBpedia type ‘Location’; or
- an *incorrectly linked mention* if it has been correctly identified as a mention, but has been incorrectly disambiguated with a KB resource which is not representative of the mention in the real world, such as the correctly identified mention *America* has been disambiguated with an incorrect KB resource http://dbpedia.org/resource/America_America of DBpedia type ‘Film’. This can be attributed to factors such as the mention is *Out of Vocabulary*

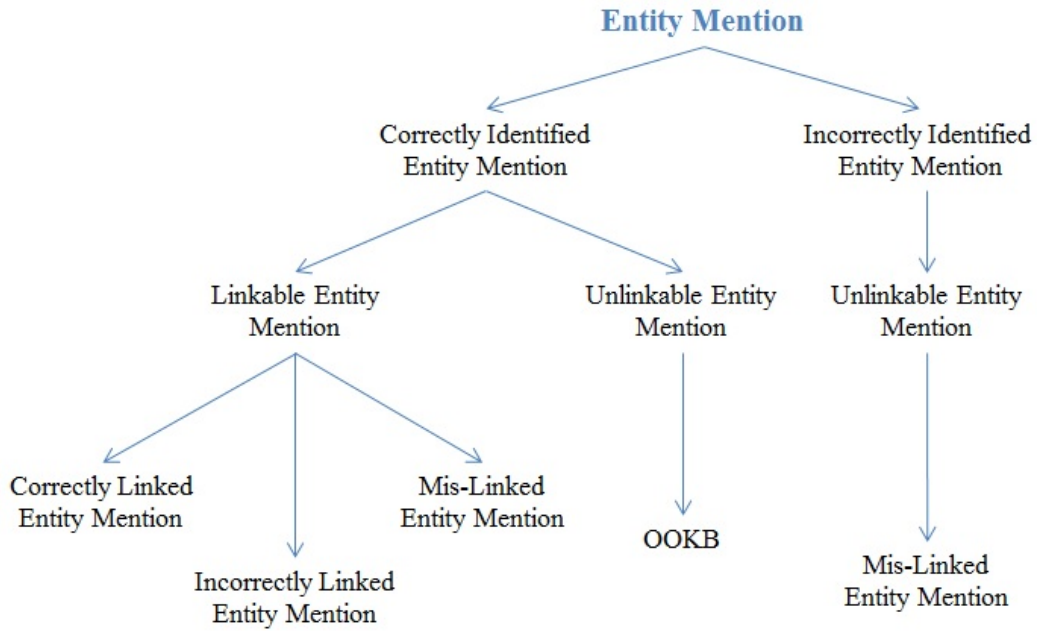


Figure 4.4: Linking Scenarios

(OOV) (for instance, the entity mention *HUD* refers to the KB resource http://dbpedia.org/resource/United_States_Secretary_of_Housing_and_Urban_Development, however HUD is an abbreviated term often used in tweets and can pose difficulties while retrieving candidates from a KB for entity disambiguation) or simply because the linking algorithm is not able to gather enough evidence to disambiguate a given mention with a suitable candidate resource; or

- a *mis-linked mention* if it has been correctly identified as a mention, but is either too generic, polysemous or synonymous with other resources in the KB and can thus lead to mislinking, such as the identified mention *N.J.* or *Steve* which are too generic and thus lead to retrieval of uninformative candidate resources which further pose difficulties for entity disambiguation. Polysemy and synonymy have been discussed previously in Section 2.3.1.

As evident from Table 4.4, 63.12% of linkable mentions have been correctly linked, whereas a small percentage of linkable mentions are incorrectly linked (3.05%) and a rather bigger percentage is mis-linked (16.15%). Additionally,

an unlinkable mention can be:

- a *mis-linked mention* if it refers to a text phrase which has been incorrectly identified as an entity mention (although, theoretically, it should not have been) by the NER system but is still able to generate candidate matches from the KB. In such cases, the candidate matches are mostly uninformative thus leading to wrongful entity disambiguation for such mentions where, more often than not, the candidate resources have been found to be associated with the DBpedia type ‘Thing’. For instance, the entity mention *Secrets* has been disambiguated with <http://dbpedia.org/resource/Secrets>; or
- a *Out of Knowledge Base mention* (OOKB) if it has been correctly identified by the NER system, but it has not (yet) been indexed by the KB since it refers to a named entity in the real world which is either newly emerging or is not relevant or popular enough so as to be indexed by the KB, such as the entity mention *Widro*.

For the sake of simplicity, a graphical representation of all the aforementioned linking scenarios is shown in Figure 4.4. A considerable fraction of unlinkable mentions are seen to be mis-linked (14.07%), i.e., have been linked to candidate resources with the parent DBpedia type ‘Thing’, Lastly, a small percentage of unlinkable mentions (3.61%) exist for which no candidate resources could be retrieved and thus, they could not be linked to any resource in the KB.

Named Entity Linking: The entity mentions linked with candidate resources that are finally selected in the previous step based on their (normalized) KB scores. The linking performance of the proposed NEEL framework is presented in Table 4.4, as also discussed in the previous section.

Entity Mention Re-classification: The entity mentions (with entity types as assigned by T-NER) that have been linked (in the linking step in the previous section) are now re-classified by using equation (3.9), irrespective of their T-NER entity types. This means that, if an entity mention e_j

Table 4.5: Comparative Analysis: T-NER and T-NER+.

Entity Type	T-NER Performance Analysis			T-NER+ Performance Analysis		
	Precision	Recall	F-Measure	Precision	Recall	F-Measure
Band	0.26	0.88	0.40	0.39	0.90	0.54
Company	0.78	0.90	0.84	0.81	0.90	0.85
Facility	0.45	0.72	0.55	0.50	0.72	0.59
Geo-Location	0.80	0.95	0.87	0.80	0.95	0.87
Movie	0.24	0.88	0.38	0.34	0.88	0.49
Other	0.57	0.70	0.63	0.56	0.76	0.64
Person	0.72	0.92	0.81	0.77	0.92	0.84
Product	0.60	0.69	0.65	0.63	0.71	0.67
Sportsteam	0.52	0.83	0.64	0.63	0.85	0.72
TVshow	0.51	0.91	0.66	0.45	0.89	0.59
Overall	0.62	0.87	0.73	0.66	0.88	0.76

with a T-NER entity type $c(e_j)$ has been disambiguated with and linked to a candidate resource r_k in DBpedia, then the mention is re-classified into the (DBpedia) type of the selected resource, as determined by $c^*(e_j)$ in equation (3.9), which may/may not be same as the original entity type as predicted by T-NER. Note that the DBpedia ontology is a highly detailed and complex ontology as compared to the Ritter ontology. Therefore, a manual mapping has been defined between the DBpedia types and T-NER types (see Table 3.1), based on which the DBpedia types of re-classified entity mentions are mapped to the T-NER types for comparative analysis. The re-classified entity types have been denoted as T-NER+ types to distinguish them with the T-NER types. This eventually, leads to improvement of the classification component of the framework as can be observed in Table 4.5 (under the column **T-NER+ Performance Analysis**).

As evident, the class-wise classification performance of majority of the entity types is improved, except for the entity type *TVshow* for which there is a decline in classification accuracy by almost 7% and the entity type *Other* with a marginal decline. Further, Table 4.6 shows some positive examples of the re-classified entity mentions that were originally *incorrectly classified* by T-NER but now have been *correctly classified* as per the entity types in the ground truth. Some negative examples of entity re-classification where

Table 4.6: Examples: Re-classification of Entity Mentions

Entity Mention	Ground-Truth Type	T-NER Type	T-NER+ Type
30stm	Band	Product	Band
Yahoo	Company	Band	Company
Southgate House	Facility	Band	Facility
Canada	Geo-Location	Person	Geo-Location
Camp rock 2	Movie	Person	Movie
Thanksgiving	Other	Person	Other
xmas	Other	TVshow	TVshow
John Acuff	Person	Facility	Person
iphone	Product	Company	Product
Lions	Sportsteam	Person	Sportsteam
TMZ	TVshow	Band	TVshow
JENNIFERS BODY	TVshow	Other	Band

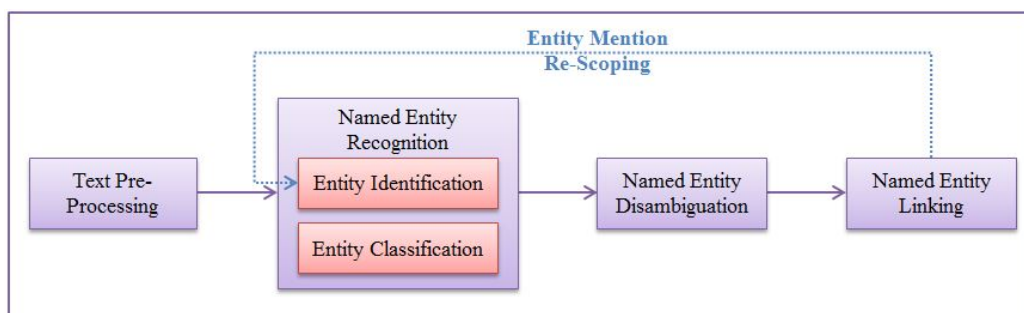


Figure 4.5: Entity Mention Re-Scoping

T-NER+ fails to correctly classify the mentions that have been incorrectly classified by T-NER are also shown in this table.

4.1.2 Entity Mention Re-Scoping

This section presents an experimental analysis of the *entity mention re-scoping* approach (discussed under Linking2Adapt in Section 3.2.2), which has been proposed in order to improve the performance of entity identification by *re-scoping* the boundaries of identified entity mentions based on the candidate resources to which they are finally linked. A schematic view of this approach is shown in Figure 4.5.

Two different datasets, i.e., the #Microposts2016 training and dev tweet

Table 4.7: Type-Wise Distribution (%): Training & Dev Ground Truth

Entity Types	Training	Dev
Character	0.73	5.62
Event	5.56	2.07
Location	21.56	5.03
Organization	18.94	9.76
Person	32.84	35.5
Product	13.84	37.87
Thing	6.58	4.14

gold standards [107], have been used for the evaluation of this approach. These gold standards, consisting of ≈ 6000 and 100 tweets, have a total of 8665 and 338 entity mentions, respectively which serve as the *training* and *dev ground truth* for the experimental analysis. The entity mentions have been classified into entity types, which constitute the Microposts Ontology, as stated in the previous chapter. A type-wise distribution of entity mentions in the training and dev ground truths is shown in Table 4.7 (as provided by the #Microposts2016 workshop organizers [107]). It can be observed that the entity types *Character* and *Event* constitute a very small percentage share in the overall distribution.

T-NER has been used again for entity identification (as in the previous section) from the training and dev datasets, while the entity types assigned by T-NER are disregarded (as stated in Section 3.2.2). A total of 8823 and 342 mentions were identified by T-NER from these ground truths respectively, out of which 4985 and 110 mentions were *correctly identified* (as per the ground truth). The identification performance of T-NER for these mentions is shown in rows labelled *without Re-scoping* in Table 4.8 for the training and dev datasets. As evident, significant precision values are obtained on both the datasets, however, recall as well as F_1 scores on the dev dataset are poor. A possible reason could be attributed to the presence of a lot of named entities present as #hashtags and @usernames in the dev ground truth, which leads to a poor performance of the entity identification component, even if @ and # are removed while pre-processing of datasets.

Table 4.8: Performance: Entity Identification

	Dataset	Precision	Recall	F-Measure
Training	without Re-scoping	0.627	0.362	0.459
	with Re-scoping	0.625	0.347	0.446
Dev	without Re-scoping	0.514	0.166	0.251
	with Re-scoping	0.545	0.178	0.268

Note that the same look-up index that was created using DBpedia in Section 4.1 has been used here as well. Further, candidate matches are retrieved from the KB for all the identified mentions (correct or incorrect), where a KB score $KB(e_j, r_k)$ is computed for each retrieved candidate using equation (3.2). As seen in Section 3.2.2, the disambiguation and linking algorithm considers features such as a candidate’s popularity in the KB into account. With the help of the linking method stated in Section 3.2.2, an entity mention is linked with a candidate resource based on its KB score. The performance of the linking algorithm is presented in rows labelled *without Re-scoping* in Table 4.9 under the column SLM² (i.e., Strong Link Match). An overall low linking performance could be attributed to poor performance of the identification component, as illustrated in Table 4.8. Further, the mentions are now classified using the DBpedia types of the candidate resources they have been linked with. Similar to the previous section, a manual mapping has been defined between the types of the DBpedia Ontology and the Microposts Ontology, based on which the mentions that have been classified using the DBpedia types are now mapped to types in the Microposts Ontology. The classification performance of this approach is shown in rows labelled *without Re-scoping* in Table 4.9 under the column STMM³ (i.e., Strong Typed Mention Match).

Post the entity linking and classification step, an entity boundary re-scoping step has been introduced in a way such that performance of entity

²It is the micro average F_1 score for a mention which has been linked to the correct candidate resource in the KB [107].

³It evaluates the micro average F_1 score for all mentions considering a mention’s boundaries and its type [107].

Table 4.9: Performance: Entity Linking and Classification

Dataset		SLM	STMM
Training	without Re-scoping	0.327	0.297
	with Re-scoping	0.336	0.300
Dev	without Re-scoping	0.194	0.139
	with Re-scoping	0.221	0.134

identification can be improved using the linking component in the proposed NEEL framework. This means that an identified mention’s boundary is re-scoped based on the candidate resource it has been linked with, if the label of the resource is a sub-string of the entity mention (as discussed in Section 3.2.2). The identification, classification (STMM) and the linking (SLM) performances of the framework for the re-scoped mentions are shown in Tables 4.8 and 4.9 respectively, in rows labelled *with Re-scoping*. As evident, the performance of the linking component improves when entity mention re-scoping is applied, for both the datasets. Additionally, the performance of the classification component improves as well for the training dataset with mention re-scoping, although, not significantly. An important observation from Table 4.8 is that by applying mention re-scoping, precision and recall scores (for entity identification) fall for the training dataset, however, its the opposite for the dev dataset. This can again be attributed to the presence of lot of #hashtags and @usernames in the dev dataset, due to which the entity identification component suffers through segmentation issues, however, mention re-scoping brings about an improvement for such cases.

Finally, Table 4.10 summarizes the performance of the linking component in terms of precision, recall and F_1 scores assuming a NER Oracle (i.e., a perfect entity recognition system). For this purpose, a modified version of the Training and Dev ground truths, denoted as Training* and Dev* have been used which comprise of linkable mentions only, i.e., void of NIL mentions. They are annotated with 6371 and 253 linkable entities, respectively. Now, based on the linking algorithm, the framework is able to link correctly $\approx 50\%$ of the mentions in the modified ground truth. However, when a NER Oracle is used, the performance of the linking component obviously falls for entity

Table 4.10: NER Oracle: Entity Linking Performance

Dataset	Precision	Recall	F ₁ Measure
Training*	0.524	0.459	0.489
Dev*	0.452	0.387	0.417

mention re-scoping. Hence, only the results without mention re-scoping have been reported for the Training* and Dev* datasets in Table 4.10.

4.2 Experimental Analysis: Learning2Adapt

In this section, an evaluation of the effectiveness of the proposed Learning2Adapt approach is presented by measuring the performance in terms of accuracy improvement for mis-classified (or incorrectly classified) entity mentions as well as mentions with uncertain entity types (as discussed in Section 3.3) with respect to two baseline approaches that exploit a deterministic manual mapping and a non-deterministic one between the source and the target domains, as well as with the NEEL framework proposed in Section 3.1. To reiterate, Learning2Adapt is a supervised approach, which has been proposed to adapt a named entity classifier based on a source ontology to different ontologies (used by different named entity classifiers).

4.2.1 Experimental Settings

To perform an experimental analysis of the proposed approach, two benchmark datasets of tweets made available for the #Microposts2015 [104] and #Microposts2016 [107] workshops in the NEEL2015 and NEEL2016 challenges respectively have been considered as **Ground Truth (GT)**. These datasets comprise of *Train*, *Dev* and *Test* sets which have used for the experiments. Table 4.11 summarizes these datasets w.r.t the number of tweets available, and the entity mentions found in the Ground Truth. Further, **T-NER** [102] has been used to identify entity mentions from the afore-mentioned datasets, which is trained using an underlying source ontology O_S (known as the Ritter Ontology, as stated in the previous

Table 4.11: Dataset Statistics

	#Microposts 2015			#Microposts2016		
	Train	Dev	Test	Train	Dev	Test
#Tweets	3498	500	2027	6025	100	3164
#Entity Mentions (GT)	4016	790	3860	8665	338	1022
#Entity Mentions (T-NER)	2535	285	1576	4394	43	1641
Gold Standard Mentions	1660	208	1133	3003	26	98

chapter) to finally derive a NER model R_S .

Ritter Ontology (O_S): *Band, Company, Facility, Geo-Location, Movie, Other, Person, Product, Sportsteam, TVshow.*

The number of mentions recognized by T-NER have also been reported in Table 4.11. Once the entity types are recognized by R_S and classified according to O_S , they need to be mapped to the entity types available in a target ontology. For the experimental analysis of the said approach, the Microposts Ontology, O_T has been used as the target ontology, as indicated in the previous chapter as well.

Microposts Ontology (O_T): *Character, Event, Location, Person, Product, Organization, Thing.*

In order to create the input space for the proposed Learning2Adapt model, a training set needs to be created, where for each entity mention identified by T-NER the probability distribution $P(\Omega_t, O_S)$, the source type $c_S \in O_S$ and the target type $c_T \in O_T$ have to be derived. While the probability distribution and the source type are explicitly provided by the T-NER system, the target type needs to be specified. However, when selecting a target type one should take into account that an entity mention recognized by T-NER could be wrongly segmented, where some tokens of a multi-word entity mention can be identified as non-mention or a single-word entity mention can be coupled with some adjoining words and therefore wrongly segmented as a multi-word mention. Two examples are reported as

below.

The **[Empire State]**_{Geo} **[Building]**_{Other} is amazing!

[Bob Sinclar will]_{Pers} be in Rome next week!

To finally induce the Learning2Adapt model, a **Gold Standard (GS)** was therefore created for the *Train* datasets of #Microposts2015 and #Microposts2016, which consists of couples $\langle e_j, c_T \rangle$ where e_j is an entity mention identified by the given T-NER system and c_T is the correct type for that mention in the target domain. For each tweet, each entity mention e_j (*T-NER*) identified by T-NER is associated with the most similar entity e_i (*GT*) in the Ground Truth. A couple $\langle e_j, c_T \rangle$ is added to the Gold Standard if and only if there is a perfect match between the entity mentions e_j (*T-NER*) and e_i (*GT*). Some examples of couples included in the Gold Standard are reported in Figure 4.6. Similar gold standards have also been created for the

Tweet ID	Mention t_i (T-NER)	T-NER source Type (c_i)	Mention t_i (GT)	Mapping Score	Ontology target Type (c_i)	Gold Standard
w ₁	Ladbroke Grove Sainsbury	Facility	Ladbroke Grove	0.92	--	✗
w ₁	Ladbroke Grove Sainsbury	Facility	Sainsbury's	0.51	Organization	✗
w ₂	Mark Twain	Movie	Mark Twain	1.00	Person	✓
w ₃	Ron Weasley	Band	Ron Weasley	1.00	Character	✓

Figure 4.6: Example of entity mentions in the Gold Standard

Dev and *Test* datasets. The number of mentions in the gold standards of the train, dev and test datasets have been reported in Table 4.11. Further, in order to compare the proposed approach with a reference, two Baseline models for re-classifying entity mentions have been defined:

- **Baseline-Deterministic (BL-D)**: manual mappings between O_S and O_T have been considered to generate this baseline, as shown in Figure 3.4;
- **Baseline-Probabilistic (BL-P)**: the afore-mentioned baseline has been extended in order to deal with fork mappings in a non-deterministic way. In particular, for those mentions in O_S that can

Table 4.12: GS Type Distribution (%) according to Source Ontology (W.C.)

	#Microposts 2015			#Microposts2016		
	Train	Dev	Test	Train	Dev	Test
Band	2.89	2.88	2.56	2.83	0.00	0.00
Company	10.78	6.73	4.68	8.29	7.69	3.06
Facility	1.93	2.40	4.15	2.63	0.00	1.02
Geo-Loc	32.29	28.85	43.51	36.13	26.92	13.27
Movie	1.45	0.96	1.32	1.43	0.00	5.10
Other	8.67	12.02	8.38	9.06	11.54	13.27
Person	37.17	43.27	32.74	35.76	46.15	50.00
Product	1.87	1.44	1.06	1.57	3.85	12.24
Sportsteam	2.29	0.96	0.97	1.67	0.00	0.00
TVshow	0.66	0.48	0.62	0.63	3.85	2.04

be classified in more than one type in O_T , the target type has been sampled according to an apriori distribution of mapping in the training set (for example, for entity mentions classified as type *Person* in O_S , 30% of them have been mapped to the type *Character* and the remaining 70% as type *Person* in O_T). This baseline has been inspired by the work presented in [2].

In addition to comparing the results with the baselines, the results are also compared with the NEEL framework proposed in Section 3.2.2, which now can be considered as a state-of-the-art approach for this experimental analysis, thus hereafter, referred to as **Caliano et al.** [18]. Regarding Learning2Adapt, the input space used for training the models has been derived using the Ritter system [102] (T-NER). In particular, LabelledLDA [96] in T-NER has been used to derive $P(\Omega_T, O_S)$ for the subsequent Learning2Adapt training phase.

As seen in Chapter 2, various state-of-the-art systems use different kinds of contextual information available from the text and/or the KB under consideration for entity recognition and entity linking. Therefore, as discussed in Section 3.3.4, the input space for the mentions in the gold standards has

Table 4.13: GS Type Distribution (%) according to Source Ontology (Wo.C.)

	#Microposts 2015			#Microposts2016		
	Train	Dev	Test	Train	Dev	Test
Band	3.19	2.40	3.18	3.26	0.00	3.06
Company	8.86	5.77	4.24	6.99	11.54	4.08
Facility	1.99	2.40	3.00	2.53	0.00	2.04
Geo-Loc	28.86	27.88	39.89	33.17	23.08	9.18
Movie	1.87	0.48	1.59	1.86	3.85	3.06
Other	11.93	15.87	12.62	12.32	3.85	18.37
Person	35.24	40.38	30.98	33.97	46.15	43.88
Product	3.67	1.44	1.85	2.86	3.85	12.24
Sportsteam	3.07	1.92	1.24	2.16	0.00	0.00
TVshow	1.33	1.44	1.41	0.87	7.69	4.08

Table 4.14: GS Type Distribution according to Target Ontology (%)

	#Microposts 2015			#Microposts2016		
	Train	Dev	Test	Train	Dev	Test
Character	1.27	0.96	0.62	1.00	11.54	29.59
Event	1.75	12.02	5.38	3.83	0.00	3.06
Location	30.60	32.21	48.81	37.63	23.08	15.31
Organization	24.82	12.98	13.77	19.85	11.54	7.14
Person	31.69	35.58	25.24	29.57	50.00	23.47
Product	7.53	3.85	3.80	5.83	3.85	15.31
Thing	2.35	2.40	2.38	2.30	0.00	6.12

been created by using two different settings: *With Context, W.C.* and *Without Context, Wo.C.*. The type distributions of all the entity mentions in the gold standards as per the source ontology while taking into account the contextual information surrounding an identified entity mention in a tweet (i.e., presence of context – W.C. setting) is provided in Table 4.12 and the type distribution while no contextual information from a tweet is taken into account (i.e., absence of context – Wo.C. setting), as explained in Section 3.3.4 is provided in Table 4.13. As observed from Tables 4.12 and 4.13, the type distributions for entity mentions vary when contextual information is considered (i.e., in the W.C. setting) and when contextual information is not considered (i.e., Wo.C. setting) since the entity types of the entity mentions are determined based on equations (3.21) and (3.22) described in Section 3.3.4 for W.C. and Wo.C. settings respectively. Finally, Table 4.14 presents the type distributions of the entity mentions in the gold standards based on the target ontology.

To compare Learning2Adapt with the baseline model and Caliano et al. both for #Microposts2015 and #Microposts2016, several performance measures have been considered. In particular, Precision, Recall, (macro-averaged) F-Measure and Strong Typed Mention Match (STMM), which corresponds to the (micro-averaged) F-Measure, have been used for comparing the types predicted by Learning2Adapt for the entity mentions with the real types available in the Ground Truth.

4.2.2 Experimental Evaluations

In this section, a detailed analysis of Learning2Adapt has been provided by taking into account the afore-mentioned settings, i.e., W.C. and Wo.C. for the training, dev and test datasets using various learning models that have been described in Section 3.3.3. A 10-fold cross validation has been used for the *training* dataset, while results for the *dev* and *test* datasets have been reported by training the models on the *training* dataset. Further, the capabilities of Learning2Adapt when it comes to dealing with the three issues stated in Section 3.3.2, i.e. mention misclassification, type uncertainty and fork mapping have been measured. To this end, an analysis (for W.C. and

Table 4.15: Learning2Adapt Capabilities: Train 2015 & Train 2016

Learning Models	Gold Standard - Train 2015						Gold Standard - Train 2016					
	MMCM (%)		TUCM (%)		FMCR (%)		MMCM (%)		TUCM (%)		FMCR (%)	
	W.C.	Wo. C.	W.C.	Wo. C.	W.C.	Wo. C.	W.C.	Wo. C.	W.C.	Wo. C.	W.C.	Wo. C.
BN	14.95	22.71	28.26	27.12	40.00	40.00	22.02	27.14	45.51	38.53	63.45	46.90
DT	27.03	36.07	47.06	45.76	12.00	2.00	33.81	40.50	51.50	56.88	44.83	47.59
KNN	27.69	34.92	42.35	45.76	8.00	18.00	33.81	39.52	51.50	52.29	47.59	47.59
MLR	22.86	25.95	37.65	45.76	0.00	0.00	21.90	25.73	38.32	43.12	11.72	13.79
MLP	23.74	35.88	41.18	57.63	0.00	0.00	31.29	32.03	50.90	53.21	44.14	31.72
NB	14.73	22.90	28.24	27.12	40.00	40.00	22.02	26.93	46.11	37.61	63.45	46.90
SVM	22.20	25.57	36.47	42.37	0.00	0.00	23.23	24.00	33.53	31.19	6.21	0.00
BL-D	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BL-P	0.00	0.00	0.00	0.00	0.00	24.00	0.00	0.00	0.00	0.00	38.62	34.48

Wo.C. setting) on the *Training* Gold Standards both for #Microposts2015 and #Microposts2016 in the form of the following measures has been provided in Table 4.15 w.r.t both the baselines BL-D and BL-P:

1. **Mention Misclassifications Correctly Mapped (MMCM):** This measure represents the percentage of entity mentions that T-NER system has incorrectly classified and Learning2Adapt is able to correctly map according to an entity type in the target ontology. For instance, in the Training Gold Standards for #Micropost2015 and #Micropost2016 (for the Wo.C. setting), T-NER incorrectly classifies 524 and 921 entity mentions respectively.
2. **Type Uncertainty Correctly Mapped (TUCM):** This measure represents the percentage of entity mentions that have uncertain entity types and that Learning2Adapt correctly maps to an entity type in the target ontology. To compute this measure, an entity mention e_j that has a smaller difference among probabilities of various entity types of the source ontology is considered as an *uncertain mention*. More formally, e_j is considered as *uncertain* if:

$$P(e_j, c_{T_i}) - P(e_j, c_{T_k}) \leq \alpha \quad \forall i \neq k \quad (4.4)$$

where α is a parameter that has been chosen to be equal to 0.2. For instance, 59 entity mentions in the #Micropost2015 Training Gold Standard and 109 in the #Micropost2016 Training Gold Standard have been recognized as *uncertain mentions* (for the Wo.C. setting).

3. **Fork Mappings Correctly Resolved (FMCR)**: Fork mappings are cases where an entity mention classified as an entity type in the source ontology (such as *Person*) can be mapped to more than one entity types in the target ontology (such as *Person* or *Character*) depending on the specificity or generality of the entity mention’s usage context. This measure represents the percentage of entity mentions that have been correctly mapped by Learning2Adapt to an entity type which falls under fork mappings in the target ontology. The number of mentions that fall under the category of fork mappings are 50 and 145, for Training Gold Standards of #Micropost2015 and #Micropost2016 respectively.

The first consideration that can be derived from Table 4.15 is that although (almost) all the learning models show promising performance for the said issues as compared to the baselines for the training gold standards (Train2015 and Train2016) for #Microposts2015 and #Microposts2016, in particular, it can be observed that the best performance w.r.t. capabilities of Learning2Adapt are obtained when using **Decision Trees**, which show a slight improvement over the other learning models. Secondly, it can also be observed that in most cases, the results on Train2015 set are lower than the ones on Train2016. This is due to the fact that the number of entity mentions available for training Learning2Adapt in the Train2016 are about twice as much than in Train2015. In other words, the higher the number of mentions that Learning2Adapt can use to learn the correct mappings, the better would be the capabilities of Learning2Adapt.

Further, in order to better understand the poor results of FMCR, a detailed investigation has been conducted on the predictions obtained by each model. For Train2015, the number of mentions involved in a fork mapping are 50 (21 for the entity type *Character* and 29 for the entity type *Event*). Given the low percentage distribution of these entity types in the training gold standard (note that the entity types *Location*, *Person* and *Organization* are composed of more than 400 instances each), it is very difficult for a machine learning algorithm to learn how to recognize their presence. On the other hand, in Train2016, there are 145 mentions involved in a fork mapping; of which 30 are of the type *Character* and 115 of the type *Event*.

The results in terms of FMCR are promising, however, following the previous intuition, the increase is mainly due to correctly classified instances for the entity type *Event*, since only few instances of the type *Character* have been correctly identified. However, this proves that with some additional training instances the ability of Learning2Adapt to deal with fork mappings can rapidly increase.

From Table 4.15, high results of the probabilistic baseline (BL-P) in terms of FMCR can be easily noted. For Train2016, BL-P is able to correctly map 79 out of 115 mentions of type *Event* for the Wo.C. setting and 56 out of 115 for the W.C. setting. Although this result seems promising, it should also be noted that BL-P over-classifies the *Event* type by 199 instances for the W.C. setting (6.62% of the total number of mentions in the dataset) and 262 instances (8.72% of the total number of mentions in the dataset) for the Wo.C. setting. This behavior can be also noted by observing the class-wise performance for the type *Event*, i.e., 0.23 and 0.69 in terms of Precision and Recall respectively in Table A11 (given in Appendix A.4) for the Wo.C. setting and 0.22 and 0.49 in terms of Precision and Recall respectively in Table 4.26 for the W.C. setting.

Another important observation from Table 4.15 is that the learning models show better performance for the capabilities when the input space for entity mentions without using contextual information (i.e., for the Wo.C. setting) is considered. This could possibly be because when contextual evidence from a tweet is being taken into consideration, there is always a risk of introducing noise, which can perpetually effect the type prediction of an entity type. Thus, while there might be slight improvement for the type mapping from source to target ontology as far as measures such as precision, recall and accuracy are concerned, the performance of the capabilities decrease. Therefore, the results of the Dev and Test gold standards of #Microposts2015 and #Microposts2016 (i.e., Dev2015 and Dev 2016, Test2015 and Test2016) for the Wo.C. setting, which have been obtained by training the models on their corresponding Training gold standards, have been provided in Table 4.16⁴.

⁴The results for the W.C. setting have been provided in Table A1 in Appendix A.1.

Table 4.16: Learning2Adapt Capabilities : Dev, Test 2015 & Dev, Test 2016 (Wo.C.)

Learning Models	Gold Standard 2015						Gold Standard 2016					
	MMCM (%)		TUCM (%)		FMCR (%)		MMCM (%)		TUCM (%)		FMCR (%)	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
BN	22.64	25.93	25.00	25.81	22.22	22.06	33.33	18.92	100.00	28.57	0.00	3.13
DT	28.30	35.04	12.50	32.26	11.11	2.94	33.33	32.43	100.00	42.86	0.00	3.13
KNN	28.30	37.32	12.50	27.42	11.11	5.88	33.33	32.43	100.00	42.86	0.00	9.38
MLR	24.53	23.08	12.50	24.19	0.00	0.00	16.67	18.92	0.00	42.86	0.00	0.00
MLP	33.96	37.32	25.00	33.87	0.00	0.00	33.33	24.32	100.00	42.86	0.00	3.13
NB	22.64	33.62	25.00	30.65	22.22	2.94	16.67	18.92	0.00	28.57	0.00	3.13
SVM	26.42	23.08	25.00	24.19	0.00	0.00	16.67	18.92	0.00	28.57	0.00	0.00
BL-D	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BL-P	0.00	0.00	0.00	0.00	74.07	61.76	0.00	0.00	0.00	0.00	0.00	0.00

It can be observed from these tables that the models perform quite well for the MMCM and TUCM capabilities, outperforming the baselines, whereas the probabilistic baseline BL-P outperforms the models for the FMCR results for the dev and test datasets for both the settings for #Microposts2015. This is so because, the number of mentions involved in a fork mapping in the Dev2015 dataset are 27 (2 for the entity type *Character* and 25 for the entity type *Event*). While none of the *Character* types get correctly mapped due to an extremely low number of this entity type in the dataset ($\approx 1\%$ of the total mentions in the dev dataset), 80% of the mentions involved in the fork mapping with type *Event* get correctly mapped since most of the entity mentions are repetitive. Similar reasoning also goes for the Test2015, where 68 mentions are involved in a fork mapping in the Wo.C. setting (7 for the entity type *Character* and 61 for the entity type *Event*). While none of the *Character* types get correctly mapped due to an extremely low number of this entity type in the dataset (0.61% of the total mentions in the test dataset), 68% of the mentions involved in the fork mapping with type *Event* get correctly mapped since most of the entity mentions are repetitive in this dataset as well. Moreover, the FMCR results are quite poor for the Dev2016 and Test2016 as compared to #Microposts2015 ones since the Dev2016 and Test2016 have very few entity mentions (see Table 4.11). Lesser the number of mentions, more difficult it is for a learning model to recognize and learn the presence of fork mappings.

A comparative analysis of Learning2Adapt with the Baseline Models and the state-of-the-art approach (Caliano et al. [18]) in terms of Precision, Recall, Weighted F-Measure and Strong Type Mention Measure (STMM) for Train2015 and Train2016 considering both W.C. and Wo.C. settings have been provided in Table 4.17 and Table 4.18 respectively.

Table 4.17: Precision, Recall, F-Measure & STMM: Train2015

	BL-D		BL-P		Caliano et al		DT		NB		SVM		BN		MLP		MLR		KNN	
	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.
Precision	0.71	0.67	0.71	0.67	0.58	0.58	0.72	0.71	0.72	0.75	0.70	0.69	0.72	0.75	0.69	0.70	0.70	0.69	0.70	0.71
Recall	0.71	0.67	0.71	0.67	0.58	0.58	0.74	0.74	0.69	0.69	0.75	0.73	0.69	0.69	0.74	0.75	0.75	0.73	0.74	0.74
F-Measure	0.71	0.67	0.71	0.67	0.58	0.58	0.73	0.72	0.69	0.70	0.72	0.70	0.69	0.70	0.71	0.73	0.72	0.71	0.71	0.72
STMM	0.40	0.38	0.38	0.37	0.35	0.35	0.43	0.40	0.36	0.39	0.39	0.38	0.36	0.38	0.39	0.40	0.39	0.38	0.40	0.43

Table 4.18: Precision, Recall, F-Measure & STMM: Train2016

	BL-D		BL-P		Caliano et al		DT		NB		SVM		BN		MLP		MLR		KNN	
	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.
Precision	0.70	0.66	0.71	0.69	0.61	0.61	0.75	0.75	0.75	0.73	0.73	0.68	0.75	0.73	0.75	0.72	0.71	0.71	0.74	0.75
Recall	0.70	0.66	0.71	0.69	0.61	0.61	0.77	0.77	0.73	0.72	0.75	0.72	0.73	0.72	0.77	0.75	0.75	0.73	0.77	0.77
F-Measure	0.70	0.66	0.71	0.69	0.61	0.61	0.76	0.75	0.74	0.72	0.72	0.70	0.74	0.72	0.75	0.73	0.72	0.71	0.75	0.75
STMM	0.38	0.37	0.42	0.42	0.35	0.35	0.51	0.50	0.44	0.44	0.40	0.37	0.44	0.44	0.49	0.45	0.41	0.42	0.49	0.51

It can be easily noted that Learning2Adapt outperforms both the Baselines and Caliano et al. both for Train2015 and Trains2016. This can be explained by the fact that according to the manual mappings as shown in Figure 3.4, the entity types *Band*, *Company*, *Geo-Location*, *Movie*, *Product*, *Sportsteam* and *TVshow* of the source ontology are mapped directly to the entity types *Organization*, *Organization*, *Location*, *Product*, *Product*, *Organization* and *Product* respectively of the target ontology without being involved in fork mappings. Entity mentions of these types comprise 52.23% and 52.55% of the total type distribution in Train2015 and Train2016 respectively (as shown in Table 4.12). As evident from Table 4.19 and Table 4.20, the Baseline Models are able to correctly map $\approx 71\%$ of the mentions and Caliano et al. shows an even lower performance of mapping ($\approx 61\%$ of the mentions), while Learning2Adapt is able to increase the performance accuracy upto 75% and 77% of the mentions for Train2015 and Train2016 respectively. These results also include fork mappings, and uncertain entity types such as *Person* and *Other*.

Table 4.19: Class Wise Accuracy Contribution (%): Train2015

Entity Type	BL-D		BL-P		Caliano et al		DT		NB		SVM		BN		MLP		MLR		KNN		
	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	
Character	0.00	0.00	0.00	0.00	0.48	0.48	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Event	0.00	0.00	0.00	0.00	0.30	0.30	0.36	0.06	1.20	1.20	0.00	0.00	1.20	1.20	0.00	0.00	0.00	0.00	0.00	0.24	0.54
Location	26.87	24.76	26.87	24.76	20.72	20.72	27.29	27.59	26.81	26.45	28.13	26.27	26.75	26.45	28.13	27.71	27.89	26.20	27.47	27.41	27.41
Organization	12.89	11.63	12.89	11.63	11.02	11.02	16.75	17.47	14.76	15.30	17.35	17.65	14.94	15.24	16.51	17.59	17.41	17.71	17.23	17.11	17.11
Person	28.19	27.29	28.19	27.29	22.11	22.11	27.11	26.99	25.66	25.30	27.77	26.99	25.66	25.30	26.69	27.05	27.95	27.47	26.99	26.75	26.75
Product	2.17	2.35	2.17	2.35	1.57	1.57	2.23	2.11	0.30	1.02	2.05	1.99	0.30	1.02	2.47	2.71	2.11	2.05	1.63	2.35	2.35
Thing	0.60	0.66	0.54	0.66	1.57	1.57	0.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Overall	70.72	66.69	70.66	66.69	57.77	57.77	73.86	74.22	68.73	69.28	75.30	72.89	68.86	69.22	73.80	75.06	75.36	73.43	73.55	74.16	74.16

Table 4.20: Class Wise Accuracy Contribution (%): Train2016

Entity Type	BL-D		BL-P		Caliano et al		DT		NB		SVM		BN		MLP		MLR		KNN		
	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	
Character	0.00	0.00	0.00	0.00	0.27	0.27	0.00	0.03	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Event	0.00	0.00	1.86	2.63	1.76	1.76	2.16	2.26	3.06	2.26	0.30	0.00	3.06	2.26	2.13	1.53	0.57	0.67	2.30	2.30	2.30
Location	32.23	29.77	32.23	29.77	27.34	27.34	34.27	33.63	33.07	32.93	34.30	31.90	33.03	32.93	34.27	33.97	34.23	32.13	34.20	33.83	33.83
Organization	9.52	8.72	9.52	8.72	8.23	8.23	13.12	13.32	11.72	11.89	13.39	13.62	11.75	11.92	13.62	11.92	12.82	13.22	13.29	13.39	13.39
Person	26.17	25.31	26.17	25.31	20.38	20.38	24.91	25.37	23.84	23.34	25.91	24.98	23.84	23.34	25.01	26.04	25.94	25.34	25.11	24.94	24.94
Product	1.60	2.00	1.60	2.00	1.23	1.23	1.83	1.90	1.53	1.47	1.50	1.76	1.57	1.47	1.43	1.67	1.43	1.80	1.37	1.96	1.96
Thing	0.40	0.50	0.07	0.10	1.76	1.76	0.40	0.13	0.07	0.07	0.00	0.00	0.07	0.10	0.10	0.03	0.00	0.00	0.23	0.30	0.30
Overall	69.93	66.30	71.46	68.53	60.97	60.97	76.69	76.66	73.29	71.96	75.39	72.26	73.33	72.03	76.56	75.16	74.99	73.16	76.49	76.72	76.72

According to these results, it can be asserted that the proposed Learning2Adapt approach is able to provide significant results for all the considered measures while obtaining a balanced contribution of precision and recall. Further, since the results exhibited by the models for the W.C. setting are better than the Wo.C. setting, the results for Test2015 and Test2016 for the W.C. setting are reported in Tables 4.21 and 4.22, where *Decision Tree*, *Multilayer Perceptron* and *K-Nearest Neighbor* are observed to be amongst the best performing models⁵.

Table 4.21: Precision, Recall, F-Measure & STMM: Test2015 (W.C.)

	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Precision	0.70	0.71	0.65	0.71	0.72	0.70	0.73	0.69	0.70	0.70
Recall	0.70	0.71	0.65	0.73	0.69	0.75	0.69	0.75	0.75	0.73
F-Measure	0.70	0.71	0.65	0.72	0.70	0.72	0.70	0.72	0.72	0.71
STMM	0.35	0.34	0.34	0.36	0.33	0.36	0.33	0.36	0.35	0.36

⁵The results for the test datasets for the Wo.C. setting as well as for the dev datasets for both the settings for #Microposts2015 and #Microposts2016 have been provided in Appendix A.2.

Table 4.22: Precision, Recall, F-Measure & STMM: Test2016 (W.C.)

	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Precision	0.47	0.47	0.31	0.67	0.38	0.35	0.38	0.37	0.35	0.33
Recall	0.47	0.47	0.31	0.53	0.46	0.50	0.46	0.51	0.50	0.47
F-Measure	0.47	0.47	0.31	0.44	0.41	0.41	0.41	0.42	0.40	0.38
STMM	0.35	0.35	0.23	0.44	0.38	0.34	0.38	0.40	0.34	0.36

In order to compare the performance accuracy of the proposed approach with the Baseline models, the class-wise accuracy for Training and Test datasets of #Microposts2015 and #Microposts2016 for the learning models are reported in Tables 4.19, 4.20, 4.23 and 4.24, where Decision Tree, Multilayer Perceptron and K-Nearest Neighbor are observed to be the best performing models. The performance related to the types involved in a fork mapping, i.e. from type *Person* in the source ontology to types *Person* and *Character* in the target ontology, as well as from *Other* to *Thing* and *Event*, are significant when compared to the Deterministic Baseline. This is mainly due to the fact that BL-D is not able (by definition) to map any *Person* to *Character* and any *Other* to *Event*. The Probabilistic Baseline is the best model for classifying entities of type *Event*. However, this result has a drawback in the sense of an over-classification of instances.

Table 4.23: Class Wise Accuracy Contribution (%): Test2015 (W.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Event	0.00	2.03	2.91	0.26	1.32	0.00	1.32	0.00	0.00	0.18
Location	41.48	41.48	37.51	44.22	42.01	44.48	42.01	44.84	43.60	43.60
Organization	4.77	4.77	4.41	7.77	6.09	8.03	6.35	6.53	8.30	7.86
Person	22.07	22.07	16.95	20.12	19.15	21.36	19.15	21.80	21.62	20.21
Product	1.06	1.06	0.88	0.97	0.18	1.06	0.18	1.59	1.06	1.06
Thing	0.18	0.00	2.12	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Overall	69.55	71.40	64.78	73.35	68.76	74.93	69.02	74.76	74.58	72.90

Table 4.24: Class Wise Accuracy Contribution (%): Test2016 (W.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00	0.00	0.00	1.02	0.00	0.00	0.00	0.00	0.00	0.00
Event	0.00	0.00	1.02	1.02	1.02	0.00	1.02	1.02	0.00	1.02
Location	10.20	10.20	8.16	13.27	11.22	13.27	11.22	12.24	12.24	13.27
Organization	2.04	2.04	0.00	4.08	2.04	2.04	2.04	2.04	2.04	3.06
Person	21.43	21.43	15.31	20.41	17.35	21.43	0.00	21.43	21.43	19.39
Product	13.27	13.27	1.02	13.27	13.27	13.27	13.27	14.29	14.29	10.20
Thing	0.00	0.00	5.10	0.00	1.02	0.00	1.02	0.00	0.00	0.00
Overall	46.94	46.94	30.61	53.06	45.92	50.00	28.57	51.02	50.00	46.94

Moreover, as evident from Table 4.19, the Baselines, Caliano et al., as well as the models show poor performance for the entity types *Character*, *Event* and *Thing*. This is so because these entity types constitute 1.26%, 1.74% and 2.35% respectively of the type distribution in the Train2015 dataset and 0.62%, 5.38% and 2.38% respectively in the Test2015 dataset (see Table 4.14). Lesser the number of mentions, harder it is for a learning model to learn and perform, as also stated previously. Similar reasoning can be provided for the performance of the models on the Test2016 dataset in Table 4.24, since as per the type distribution in the Train2016 dataset according to the Target Ontology, the entity types *Character*, *Event* and *Thing* comprise of 1.00%, 3.83% and 2.30% respectively while their percentage distribution in the Test2016 is 29.59%, 3.06% and 6.12% respectively (see Table 4.14). Although, the type *Character* constitutes of $\approx 30\%$ of the type distribution in Test2016, the corresponding training set does not have enough training instances for the model to be able to learn and perform accordingly⁶.

⁶The class-wise accuracy for Dev2015 and Dev2016 for both W.C. and Wo.C. settings and for Test2015 and Test2016 for the Wo.C. setting are reported in Appendix A.3.

Table 4.25: Class Wise Precision / Recall: Train2015 (W.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.80/0.38	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.00/0.00	0.56/0.17	0.23/0.21	0.08/ 0.69	0.00/0.00	0.08/ 0.69	0.00/0.00	0.00/0.00	0.12/0.14
Location	0.83/0.88	0.83/0.88	0.79/0.68	0.80/0.89	0.81/0.88	0.81/ 0.92	0.81/0.87	0.82/ 0.92	0.82/0.91	0.79/0.90
Organization	0.81/0.52	0.81/0.52	0.79/0.44	0.69/0.67	0.68/0.59	0.66/ 0.70	0.69/0.60	0.64/0.67	0.67/ 0.70	0.67/0.69
Person	0.76/ 0.89	0.76/ 0.89	0.94/0.70	0.81/0.86	0.86/0.81	0.79/0.88	0.86/0.81	0.80/0.84	0.78/0.88	0.81/0.85
Product	0.55s/0.29	0.55/0.29	0.28/0.21	0.47/0.30	0.42/0.04	0.55/0.27	0.38/0.04	0.43/ 0.33	0.52/0.28	0.48/0.22
Thing	0.06/0.26	0.05/0.23	0.05/ 0.67	0.18/0.05	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00

Table 4.26: Class Wise Precision / Recall: Train2016 (W.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.57/0.27	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.22/0.49	0.87/0.46	0.63/0.57	0.25/ 0.80	0.00/0.08	0.25/ 0.80	0.84/0.56	0.00/0.15	0.64/0.60
Location	0.89/0.86	0.89/0.86	0.86/0.73	0.85/0.91	0.88/0.88	0.86/ 0.91	0.88/0.88	0.86/ 0.91	0.86/ 0.91	0.85/ 0.91
Organization	0.74/0.48	0.74/0.48	0.72/0.41	0.68/0.66	0.71/0.59	0.58/0.67	0.70/0.59	0.61/ 0.69	0.61/0.65	0.64/0.67
Person	0.73/ 0.89	0.73/ 0.89	0.92/0.69	0.80/0.84	0.84/0.81	0.77/0.88	0.84/0.81	0.78/0.85	0.75/0.88	0.79/0.85
Product	0.44/0.27	0.44/0.27	0.26/0.21	0.42/ 0.31	0.39/0.26	0.50/0.26	0.39/0.27	0.50/0.25	0.45/0.25	0.53/0.23
Thing	0.03/0.17	0.02/0.03	0.06/ 0.77	0.32/0.17	0.05/0.03	0.00/0.00	0.05/0.03	0.38/0.04	0.00/0.00	0.32/0.10

Table 4.27: Class Wise Precision / Recall: Test2015 (W.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.26/0.38	0.89/0.54	0.14/0.05	0.09/0.25	0.00/0.00	0.09/0.25	0.00/0.00	0.00/0.00	0.11/0.03
Location	0.95/0.85	0.95/0.85	0.92/0.77	0.89/0.91	0.89/0.86	0.92/0.91	0.94/0.86	0.91/ 0.92	0.93/0.89	0.89/0.89
Organization	0.58/0.35	0.58/0.35	0.55/0.32	0.46/0.56	0.48/0.44	0.41/0.58	0.42/0.46	0.48/0.47	0.47/ 0.60	0.43/0.57
Person	0.67/ 0.87	0.67/ 0.87	0.90/0.67	0.75/0.80	0.80/0.76	0.74/0.85	0.80/0.76	0.67/0.86	0.69/0.86	0.73/0.80
Product	0.35/0.28	0.35/0.28	0.26/0.23	0.25/0.26	0.29/0.05	0.35/0.28	0.25/0.05	0.34/ 0.42	0.27/0.28	0.38/0.28
Thing	0.01/0.07	0.00/0.00	0.08/0.89	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00

Table 4.28: Class Wise Precision / Recall: Test2016 (W.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.00/0.00	1.00/0.03	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.00/0.00	0.08/ 0.33	1.00/0.33	0.05/ 0.33	0.00/0.00	0.05/ 0.33	0.50/ 0.33	0.00/0.00	0.25/ 0.33
Location	0.77/0.67	0.77/0.67	0.73/0.53	0.72/ 0.87	0.65/0.73	0.81/0.87	0.65/0.73	0.80/0.80	0.80/0.80	0.65/ 0.87
Organization	0.67/0.29	0.67/0.29	0.00/0.00	0.20/ 0.57	0.33/0.29	0.10/0.29	0.33/0.29	0.11/0.29	0.12/0.29	0.18/0.43
Person	0.43/ 0.91	0.43/ 0.91	0.83/0.65	0.53/0.87	0.55/0.74	0.48/ 0.91	0.00/0.00	0.47/ 0.91	0.46/ 0.91	0.46/0.83
Product	0.68/0.87	0.68/0.87	0.13/0.07	0.65/0.87	0.68/0.87	0.72/0.87	0.68/0.87	0.78/0.93	0.70/ 0.93	0.67/0.67
Thing	0.00/0.00	0.00/0.00	0.11/ 0.83	0.00/0.00	0.33/0.17	0.00/0.00	0.33/0.17	0.00/0.00	0.00/0.00	0.00/0.00

Further, the class-wise Precision and Recall for Train2015, Train2016, Test2015 and Test2016 are reported in Table 4.25 – 4.28 respectively for the

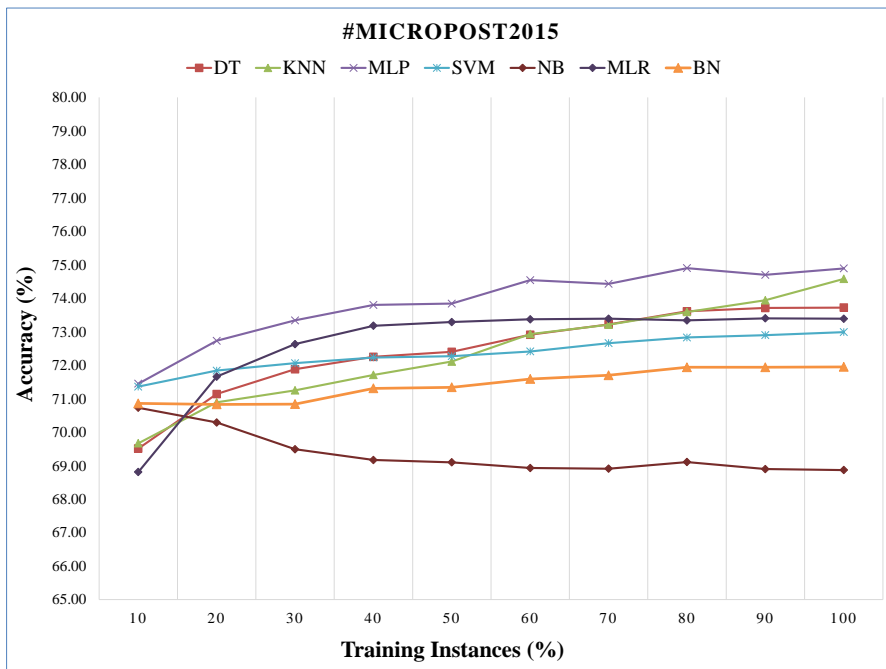
W.C. setting. An interesting insight about the Deterministic Baseline is the promising Precision achieved for the entity types *Organization* and *Location*. The main reason for the high performance for these two specific types is the strict relation with the performance of the T-NER system, as reported in Table 4.5. The (T-NER) entity types *Company* and *Geo-Location* (that are mapped to *Organization* and *Location* as per the manual mapping, in Table 3.4) are those with the highest classification performance (84% and 87% of accuracy respectively).

The results related to Probabilistic Baseline become interesting when focusing on #Microposts2016 datasets. BL-P over-estimates the distribution of the entity type *Event*, resulting in a high Recall at the expense of a low Precision. It can be seen in Table 4.28 that Decision Tree is the only model that is able to classify some mentions related to *Character* type. This is one of the most challenging types to deal with, because of the low number of instances available in the dataset and the intrinsic difficulty that a fork mapping poses. The other datasets are not able to correctly classify any mention to the type *Character*⁷.

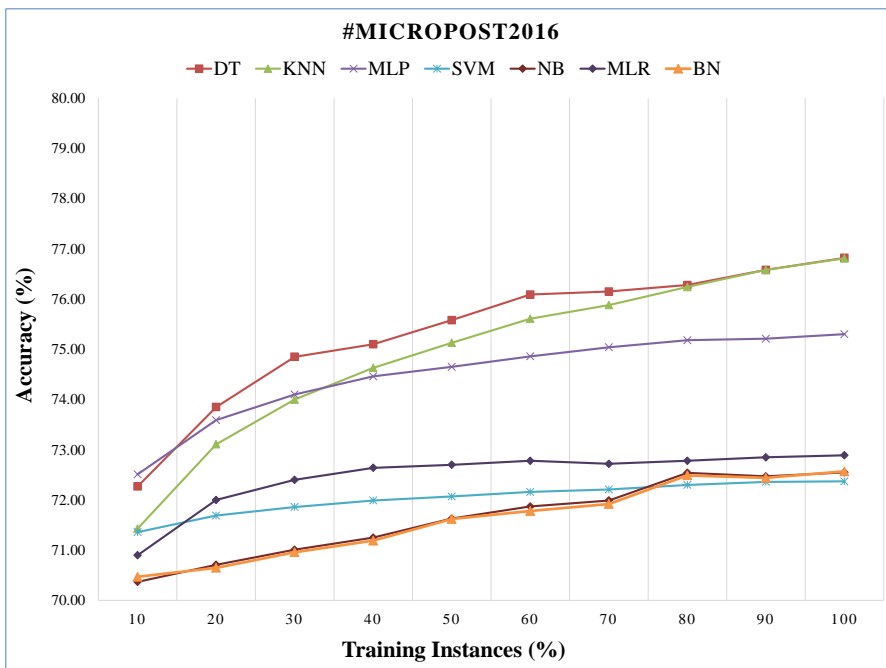
Finally, examples of entity mentions that have been correctly classified by Learning2Adapt are shown as below:

- the mention “iPhone”, which was classified into the type *Company* by T-NER (which would lead to the type *Organization* using manual mappings) has been correctly re-classified to *Product* by Learning2Adapt.
- the mention “Ron Weasley” (a character in Harry Potter), which was misclassified as *Band* by T-NER has been correctly re-classified as a *Character* by Learning2Adapt.

⁷The class-wise Precision and Recall for all the remaining datasets are reported in Appendix A.4.



(a) Train2015 Learning Curve



(b) Train2016 Learning Curve

Figure 4.7: Learning Curves

In the latter case, Learning2Adapt was able to assign the correct type among the two possible types defined according to fork mappings. Although there are very few instances in the training sets for the target types *Character* and *Event* (i.e., the types involved in fork mappings) and the performance of Learning2Adapt is not very high in this respect, the proposed approach seems to be promising.

Finally a graphical representation of the learning curves (in terms of Accuracy) for Train2015 and Train2016 is presented in Figure 4.7(a) and 4.7(b) respectively. An important observation is concerned with the ability of the proposed approach to be able to learn mappings from a source to target domain, even when a small number of training instances are available. In fact, Learning2Adapt is able to adapt an entity mention from a source to a target ontology with only 10% of the training data, ensuring an average accuracy of 71%. The only exception is represented by Naïve Bayes in #Microposts2015, where the high variance of the training data implies a model characterized by over-fitting.

4.3 Experimental Analysis: Learning2Link

This section presents an experimental analysis of the **Learning2Link** approach (**L2L**) that has been proposed in Chapter 3 Section 3.4 with the intent of improving the performance of the linking component of a NEEL framework by use of the recognition component and disambiguation component, thereby, improving the overall performance of the framework.

4.3.1 Experimental Settings

To perform an experimental analysis of the proposed approach, the same datasets of tweets: #Microposts2015 [104] and #Microposts2016 [107] that were used for the experimental analysis of Learning2Adapt, have been used as **Ground Truth** (GT) for Learning2Link as well. As mentioned in the previous section, Table 4.11 summarizes these datasets in terms of number of tweets and entity mentions available in the ground truth. Note that these datasets are annotated as per the entity types in ontology O_T , as stated

before. Similar to the previous approach, T-NER [102] (which is defined by ontology O_S , as stated before) has been used again for entity identification and classification from the datasets. Further, a NER-Oracle has also been used for an unbiased and independent evaluation of Learning2Link. The number of entity mentions identified by T-NER (as per the ontology O_S) are also reported in Table 4.11, while the entity mentions present in the ground truth serve as the NER-Oracle (as per the ontology O_T).

4.3.2 Experimental Evaluation

As described in Section 3.4, the input space for an entity mention e_j in Learning2Link is derived by using the coefficients in equation (3.2) for each candidate resource r_k that has been retrieved from the KB during entity disambiguation. These coefficients are:

- $lcs(e_j, l_{r_k})$, which denotes a normalized Lucene Conceptual Score between an entity mention e_j and the label of a candidate resource l_{r_k} ;
- $cos(e_j^*, a_{r_k})$, which represents a discounted cosine similarity between an entity context e_j^* (modeled as a vector composed of an identified entity mention e_j and non stop-words in a tweet t) and a candidate’s abstract description a_{r_k} available from the KB;
- *Jaro-Winkler distance* [64] between an entity mention e_j and the label of a resource l_{r_k} ;
- $R(r_k)$, which is a popularity measure of a given candidate resource r_k in the KB.

Note that these measures have been used previously to estimate the KB score $KB(e_j, c(e_j))$ using equation (3.2). Further, a fifth (boolean) component in the form of a predicted *target class* used for every candidate resource r_k is indicative of whether or not a candidate resource is suitable for the given entity mention. This input space is further used to learn the target class for a new mention e_n , given its candidate resources and the measures described above. Finally, the **Decision Criteria**, described in Section 3.4,

is used to determine if the new mention is linkable or unlinkable based on the predicted target class.

Further, a 10-fold cross validation has been used for the *training* dataset, while results for the *dev* and *test* datasets have been reported by training the models on the *training* dataset. Note that for every mention identified by T-NER or present in the NER Oracle, experimental results have been reported here by using an input space that has been created using *five* candidate resources that have been retrieved from the KB for entity disambiguation. Experimental results, where input space has been created using *ten* candidate resources for each entity mention, have been provided in Appendix B.

Table 4.29: #Microposts2015: F-Measure (5 instances)

Learning Models	Train 2015		Dev 2015		Test 2015	
	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)
BN	0.71	0.96	0.70	0.96	0.77	0.97
NB	0.57	0.95	0.47	0.95	0.67	0.96
SVM	0.54	0.96	0.44	0.96	0.66	0.97
MLP	0.67	0.96	0.71	0.96	0.76	0.97
VP	0.58	0.96	0.45	0.95	0.68	0.97
KNN	0.76	0.97	0.70	0.97	0.72	0.97
DT	0.81	0.98	0.77	0.97	0.80	0.98

Table 4.30: #Microposts2016: F-Measure (5 instances)

Learning Models	Train 2016		Dev 2016		Test 2016	
	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)
BN	0.72	0.97	0.52	0.95	0.20	0.91
NB	0.59	0.96	0.37	0.95	0.21	0.93
SVM	0.58	0.96	0.35	0.96	0.21	0.95
MLP	0.73	0.97	0.77	0.98	0.50	0.95
VP	0.61	0.96	0.39	0.96	0.22	0.95
KNN	0.79	0.98	0.64	0.97	0.46	0.94
DT	0.83	0.98	0.56	0.96	0.57	0.96

Tables 4.29 and 4.30 provide the F-Measure distinguishing between the predicted target classes *Yes* and *No* for the #Microposts2015 and #Microposts2016 datasets (training, dev and test) respectively, when T-NER has been used to identify the entity mentions. Tables 4.31 and 4.32 provide the F-Measure for these datasets where entity mentions are labelled by a NER Oracle.

Table 4.31: #Microposts2015: F-Measure (5 instances) - With Oracle

Learning Models	Train 2015		Dev 2015		Test 2015	
	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)
BN	0.71	0.95	0.62	0.95	0.69	0.96
NB	0.55	0.94	0.48	0.95	0.62	0.96
SVM	0.57	0.95	0.48	0.96	0.64	0.96
MLP	0.71	0.95	0.65	0.96	0.71	0.96
VP	0.58	0.95	0.49	0.96	0.65	0.97
KNN	0.80	0.97	0.65	0.96	0.71	0.96
DT	0.83	0.97	0.70	0.97	0.69	0.97

Table 4.32: #Microposts2016: F-Measure (5 instances) - With Oracle

Learning Models	Train 2016		Dev 2016		Test 2016	
	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)
BN	0.70	0.96	0.54	0.93	0.22	0.91
NB	0.58	0.95	0.43	0.93	0.20	0.93
SVM	0.59	0.95	0.35	0.94	0.23	0.95
MLP	0.73	0.97	0.65	0.96	0.48	0.95
VP	0.61	0.96	0.36	0.94	0.24	0.95
KNN	0.81	0.97	0.60	0.95	0.23	0.92
DT	0.84	0.98	0.63	0.95	0.44	0.94

The overall F-Measure is an average of the scores of *Yes* and *No* target classes and is comparable with the F-Measure of the unsupervised linking approach Caliano et al. [18], as provided in Table 4.33. A first observation that can be made from Tables 4.29 – 4.32 is that **Decision Tree** is the

best performing models for almost all datasets for #Microposts2015 and #Microposts2016. Secondly, results for F_1 (No) are always better than F_1 (Yes), because theoretically for an entity mention there will always be at-most one correct candidate resource in the KB. Thus, the number of candidate resources with the predicted target class ‘Yes’ will always be less than the number of candidate resources with the predicted target class ‘No’, which thereby has an effect on the F_1 scores.

Table 4.33: Performance Analysis: Caliano et al. vs L2L

Datasets	#Microposts2015			#Microposts2016		
	Caliano et al.	L2L (5)	L2L (10)	Caliano et al.	L2L (5)	L2L (10)
Train	0.58	0.89	0.89	0.61	0.91	0.90
Dev	0.63	0.87	0.83	0.54	0.87	0.82
Test	0.65	0.89	0.88	0.31	0.76	0.62

It can be observed from the summarized results in Table 4.33 that Learning2Link obtains higher performance when *five* candidate resources are considered for disambiguation and linking an entity mention, as opposed to *ten*, which is why the performance of Learning2Link on the datasets with *ten* instances is reported in Appendix B. Note that the best results as achieved by learning models have been reported in this table for Learning2Link, and the results for Caliano et al. can be seen in tables in Section 4.2.2. Further, it can be noted from this table that Learning2Link (supervised) outperforms the unsupervised approach (Caliano et al.) that has been proposed in Section 3.2, for both cases of when *five* and *ten* candidate resources are used.

Note that the results obtained by Learning2Link when entity mentions have been identified by T-NER (i.e., Tables 4.29 and 4.30) and when they have been labelled by a NER Oracle (i.e., Tables 4.31 and 4.32) are not comparable since the number of entity mentions identified by T-NER are lower and not always the same as those labelled by the Oracle (as can be noticed by the statistics provided in Table 4.11).

Based on the candidate resource that has been selected by the *Decision Criteria* of Learning2Link for an entity mention, the entity type of the given

mention is determined by the KB type (in this case, DBpedia type) of the selected resource, as discussed in Section 3.4. The DBpedia type is mapped to an entity type in O_T using the mappings that have been created manually. However, if no resource has been selected for a mention, then it is mapped to the type that has been determined in the recognition phase (given by $T - NER_{type}$).

Tables 4.34 – 4.37 report the SLM, *Strong Link Match*, measure for entity mentions when *five* candidate resources are selected, which is used to estimate the linking performance of a system in terms of correctly linked entity mentions. Two SLM measures have been used, defined as follows:

$$SLM1 = \sum_{i=1}^n \frac{TP(c_i)}{|GT|} \quad (4.5)$$

$$SLM2 = \sum_{i=1}^n \frac{TP(c_i)}{TP(c_i) + FN(c_i)} \quad (4.6)$$

where:

- $TP(c_i)$ denotes true positive or the number of mentions linked correctly for an entity type c_i belonging to an ontology,
- $FN(c_i)$ denotes the false negative or the number of mentions linked incorrectly for an entity type c_i ,
- $|GT|$ denotes the total number of entity mentions in the ground truth, and
- n represents the number of entity types in the ontology.

Thus, the measure $SLM1$ is used to calculate the linking performance for all entity types (in terms of correctly linked mentions) over the complete dataset. Further, $SLM2$ is used to determine the linking performance for all entity types (in terms of correctly linked mentions) over the total number of mentions classified in a given type. Further, performance for NIL (i.e., **unlinkable**) mentions is determined as the number of mentions correctly classified as NIL Mentions (since no candidate resource was selected for them

by the Decision Criteria) over the complete dataset. This is estimated in the way as shown below:

$$NIL = \sum_{i=1}^n \frac{TP(c_i)}{|GT|} \quad (4.7)$$

Table 4.34: #Microposts2015: SLM (5 instances)

Learning Models	Train 2015			Dev 2015			Test 2015		
	SLM1	SLM2	NIL	SLM1	SLM2	NIL	SLM1	SLM2	NIL
BN	0.28	0.75	0.71	0.23	0.71	0.77	0.27	0.80	0.76
NB	0.20	0.54	0.76	0.13	0.40	0.85	0.21	0.62	0.84
SVM	0.16	0.42	0.86	0.10	0.30	0.93	0.18	0.53	0.93
MLP	0.25	0.67	0.72	0.25	0.77	0.70	0.29	0.85	0.66
VP	0.18	0.48	0.83	0.10	0.32	0.90	0.19	0.56	0.92
KNN	0.29	0.77	0.78	0.21	0.65	0.82	0.37	0.68	0.80
DT	0.31	0.84	0.80	0.25	0.78	0.83	0.28	0.81	0.82

Table 4.35: #Microposts2016: SLM (5 instances)

Learning Models	Train 2016			Dev 2016			Test 2016		
	SLM1	SLM2	NIL	SLM1	SLM2	NIL	SLM1	SLM2	NIL
BN	0.32	0.74	0.77	0.12	0.62	0.65	0.05	0.24	0.65
NB	0.24	0.57	0.82	0.06	0.32	0.82	0.04	0.19	0.74
SVM	0.20	0.46	0.91	0.05	0.24	0.92	0.03	0.15	0.82
MLP	0.30	0.69	0.85	0.16	0.78	0.82	0.10	0.54	0.76
VP	0.22	0.51	0.87	0.06	0.28	0.91	0.03	0.16	0.80
KNN	0.34	0.79	0.83	0.12	0.62	0.80	0.10	0.54	0.69
DT	0.36	0.83	0.87	0.10	0.52	0.78	0.11	0.55	0.74

On close observation of Tables 4.34 and 4.35, it can be noted that performance of SLM1 is quite low for all the datasets for #Microposts2015 and #Microposts2016, while SLM2 and NIL measures exhibit high scores. This is so because SLM1 is an estimate for a correctly linked mention of a particular type over all the mentions in the dataset, or in other words, it does not consider the entity mentions in the ground truth that have not been

identified by T-NER. Poor identification performance have a counter-effect on this score, since if an entity mention has been incorrectly identified (i.e., it does not denote a named entity in the real world), then it will be incorrectly classified, will have noisy candidate resources in the input space and most probably would be incorrectly linked. On the other hand, the SLM1 performance exhibited by Learning2Link when a NER Oracle is considered is quite high for all datasets in #Microposts2015 in Table 4.36, since there are no identification and classification errors, prior to Learning2Link. It is interesting to note comparatively lower SLM performance for entity mentions in the dev and test datasets of #Microposts2016 (Table 4.37). This could be explained by the presence of mentions that are either difficult to link or are new, due to which noisy candidate resources are retrieved and used in the input space, which thereby effects the performance of Learning2Link. High performance has been observed for the NIL mentions, which means that the approach is suitable for correctly identifying NIL mentions⁸.

Table 4.36: #Microposts2015: SLM (5 instances) - With Oracle

Learning Models	Train 2015		Dev 2015		Test 2015	
	SLM1	NIL	SLM1	NIL	SLM1	NIL
BN	0.69	0.74	0.65	0.75	0.71	0.76
NB	0.49	0.74	0.44	0.81	0.58	0.85
SVM	0.46	0.84	0.36	0.90	0.53	0.90
MLP	0.71	0.71	0.66	0.71	0.73	0.74
VP	0.45	0.88	0.36	0.92	0.53	0.91
KNN	0.80	0.79	0.67	0.74	0.71	0.77
DT	0.81	0.84	0.66	0.82	0.63	0.85

⁸The results for SLM using *ten* candidate resources and for the NER oracle have been reported in Appendix B. Additionally, Learning2Link has also been used to evaluate the linking performance for entity mentions recognized from Italian language tweets. More details regarding the results are provided in Appendix B.1.

Table 4.37: #Microposts2016: SLM (5 instances) - With Oracle

Learning Models	Train 2016		Dev 2016		Test 2016	
	SLM1	NIL	SLM1	NIL	SLM1	NIL
BN	0.68	0.77	0.50	0.50	0.27	0.27
NB	0.53	0.80	0.24	0.24	0.24	0.19
SVM	0.47	0.89	0.23	0.23	0.15	0.17
MLP	0.67	0.85	0.54	0.54	0.21	0.54
VP	0.49	0.89	0.23	0.23	0.18	0.18
KNN	0.80	0.83	0.51	0.51	0.27	0.27
DT	0.81	0.88	0.57	0.57	0.27	0.56

A comparative analysis of Table 4.34 with 4.36, as well as Table 4.35 with 4.37 shows that the performance of Learning2Link is higher when a NER Oracle has been used as compared to when a named entity recognition system is used.

A detailed account of experimental evaluations had been presented in this chapter with respect to the approaches, i.e., Linking2Adapt, Learning2Adapt and Learning2Link, that have been proposed in this thesis as an effort towards improvement of Named Entity Extraction and Linking Frameworks, in particular for microblogging platforms. Different datasets, ontologies and linking algorithms, as well as different sources of contextual information have also been explored in this chapter to show the significance of the proposed approaches when such parameters are varied. Most of the results showed significant improvement over baseline models, thus proving the need and scope for improvement. The concluding remarks and the future directions are provided in detail in the next chapter.

5 CONCLUSION

The unprecedented amount of information available today from the Web of Documents has led to significant efforts towards extraction of relevant structured information and knowledge discovery. The intent of extracting structured information is two-fold: such information is highly comprehensive and easy to manipulate by computer applications for future purposes as well as its use in real-time for knowledge base enrichments. As a result, multiple IE and NLP frameworks exist today to address this task. However, when dealing with information evolving on microblogging platforms, the task becomes complicated. The difficulties posed by microblogging platforms and the challenges associated when extracting relevant information in a structured manner from such platforms have been discussed in this thesis.

Moreover, a Named Entity Extraction and Linking Framework has been presented in this thesis, in Chapter 3 Section 3.1, for dealing with microblogging platforms such as Twitter. The components in a NEEL framework have been presented in detail, along with the experimental analysis of different components. The main goal of this thesis has been to use the components of a NEEL framework in a way that one component can be used to improve the performance of the other, thus, improving the overall performance of the framework. As a result, several approaches have been proposed in this thesis to address this task.

First of all, an unsupervised approach named **Linking2Adapt** has been proposed in Chapter 3 Section 3.2 with the intent of improving the process of named entity identification by recognition component with the help of linking component. Further, Linking2Adapt has also been proposed to improve the process of named entity classification by the recognition component with the help of linking component. An experimental analysis for these tasks have been conducted using different tweet datasets (which are annotated using different ontologies) and is presented in Chapter 4 Section 4.1. Two important observations follow the experimental analysis of Linking2Adapt: (1) the

results obtained support the hypothesis that the accuracy and efficiency of a NEEL framework can be improved if one component is used to a means to support the other, and (2) different ontologies used by different datasets advocate the need of automating the task of adapting a named entity classifier to new ontologies, if need be.

To this end, the second supervised approach named **Learning2Adapt** has been proposed in Chapter 3 Section 3.3 with the purpose of adapting a named entity classifier to new ontologies in an automated way. For the experimental analysis presented in Chapter 4 Section 4.2, two different ontologies have been used to map entity mentions from a source ontology to a target ontology in an automated manner. Further, the use of contextual evidence from the text under consideration has been explored as an effort towards improving the performance of Learning2Adapt. The results show that the task of entity mentions from a source to a target ontology is not trivial and an automated system can outperform mappings between ontology classes that have established manually.

Finally, a supervised approach named **Learning2Link** has been proposed in Chapter 3 Section 3.4 as an effort towards improving the performance of the linking component and the NEEL framework by using the recognition and disambiguation components. The experimental results for the same have been presented in Chapter 4 Section 4.3 which not only prove the initial hypothesis but also show that Learning2Link is able to outperform the unsupervised approach Linking2Adapt that had been proposed initially.

5.1 Future Work

Several important concerns have been addressed in this thesis. However, no system is perfect and therefore, there is always a constant struggle in the research community to improve and innovate. Based on the challenges addressed in this thesis, and the goals accomplished, the future directions can be highlighted as follows:

- Currently, the task of extracting structured information from social media platforms has been addressed, in particular Twitter. There are

numerous other textual formats and sources of information out on the Unstructured Web, such as online news websites and Youtube, that can be used for extracting structured information in real-time.

- The approaches proposed for the improvement of the proposed NEEL framework need to be integrated to analyse the combined improvement that they have shown to bring about individually.
- The proposed framework has been deployed (and named as TWINE: TWeet analysis via INformation Extraction) and operates in real-time for named entity recognition and linking, however, the task of adapting this framework to new ontologies in real-time needs to be addressed.
- One of the reasons behind extracting structured information today from the Unstructured Web is enrichment of Knowledge Bases in real-time. For this, an important future work is the process of identification of *relevant* information and filtering out the *irrelevant* (or, in many cases *fake*) information. As of now, such entity mentions that are not indexed by a knowledge base are considered as unlinkable and categorized as NIL mentions. However, a mention can be unlinkable because it is either irrelevant for a knowledge base or it is a newly evolving mention. The distinction between these two tasks is an important future work so as to accomplish the task of knowledge base enrichment in real-time.

6 LIST OF PUBLICATIONS

- Manchanda, P., Nozza, D., Fersini, E., Palmonari, M. and Messina, E. Learning2Adapt: Learning to Classify Named Entities with a New Ontology in a Microblogging Environment. **To be submitted at** the Natural Language Engineering Journal.
- Manchanda, P., Fersini, E., Nozza, D., Palmonari, M., and Messina, E. Adapting Named Entity Classifiers to New Ontologies in a Microblogging Environment. **Submitted at** the IEEE/WIC/ACM International Conference on Web Intelligence, 2017.
- Nozza, D., Ristagno, F., Palmonari M., Fersini, E., Manchanda, P. and Messina, E. TWINE: A real-time system for TWEEt analysis via Information Extraction. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (Demo track), 2017.
- Manchanda, P., Fersini, E., Palmonari, M., Nozza, D., and Messina, E. Towards Adaptation of Named Entity Classification. In Proceedings of the 32nd Annual ACM Symposium on Applied Computing (2017), ACM.
- Cecchini Massimiliano, F., Fersini, E., Manchanda, P., Messina, E., Nozza, D., Palmonari, M., and Sas, C. UNIMIB@NEEL-IT : Named Entity Recognition and Linking of Italian Tweets. Proceedings of the 5th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA'16) (2016).
- Caliano, D., Fersini, E., Manchanda, P., Palmonari, M., and Messina, E. UNIMIB: Entity Linking in Tweets using Jaro-Winkler Distance, Popularity and Coherence. Preotiuc-Pietro et al.[20] (2016), 70–72.

- Manchanda, P., Fersini, E., and Palmonari, M. Leveraging Entity Linking to Enhance Entity Recognition in Microblogs. In Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), 2015 7th International Joint Conference, vol. 1, SCITEPRESS, pp. 147–155.
- Manchanda, P. Entity Linking and Knowledge Discovery in Microblogs. The ISWC 2015 Doctoral Consortium (2015), 25.
- Younus, A., Qureshi, M. A., Manchanda, P., O’Riordan, C., and Pasi, G. Utilizing Microblog Data in a Topic Modelling Framework for Scientific Articles’ Recommendation. In International Conference on Social Informatics (2014), Springer International Publishing, pp. 384–395.

APPENDICES

A Experimental Results: Learning2Adapt

The experimental results for Learning2Adapt approach are presented in this section in terms of its capabilities, the overall precision, recall, F-Measure and STMM (Strong Typed Mention Match). Further, class-wise accuracy and class-wise Precision and Recall are also provided for the #Microposts2015 and #Microposts2016 (train, dev, test) datasets using the W.C. or the Wo.C. setting (as described in Section 3.3.4).

A.1 Capabilities

The Learning2Adapt capabilities, i.e., MMCM, TUCM, and FMCR for dealing with mis-classified and uncertain mentions, as well as mentions that are involved with a fork mapping for Dev and Test datasets for #Microposts2015 and #Microposts2016 for the W.C. setting have been reported in this section. These capabilities have been defined in Section 3.3.2. **Decision Tree** is observed to be the best performing model for #Microposts2015 and #Microposts2016.

Table A1: Dev, Test 2015 & Dev, Test 2016 (W.C.)

Learning Models	Gold Standard 2015						Gold Standard 2016					
	MMCM (%)		TUCM (%)		FMCR (%)		MMCM (%)		TUCM (%)		FMCR (%)	
	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test	Dev	Test
BN	17.65	17.48	9.09	18.06	22.22	22.06	16.67	15.63	50.00	18.18	0.00	3.13
DT	27.45	24.27	18.18	25.00	0.00	4.41	16.67	31.25	50.00	36.36	33.33	6.25
KNN	27.45	26.21	18.18	25.00	3.70	2.94	16.67	15.63	50.00	9.09	0.00	3.13
MLR	23.53	21.36	9.09	25.00	0.00	0.00	33.33	12.50	100.00	18.18	0.00	0.00
MLP	27.45	22.01	9.09	25.00	0.00	0.00	16.67	12.50	50.00	18.18	0.00	3.13
NB	17.65	16.83	9.09	18.06	22.22	22.06	16.67	15.63	50.00	18.18	0.00	3.13
SVM	23.53	20.71	9.09	23.61	0.00	0.00	16.67	12.50	50.00	9.09	0.00	0.00
BL-D	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
BL-P	0.00	0.00	0.00	0.00	59.26	33.82	0.00	0.00	0.00	0.00	0.00	0.00

A.2 Precision, Recall, F-Measure & STMM

This section presents the experimental results in terms of Precision, Recall, F-Measure and STMM (Strong Typed Mention Match) for the Dev (W.C. and Wo.C.) and Test (Wo.C.) datasets of #Microposts2015 and #Microposts2016. The remaining results have been reported in Section 3.3. A comparative analysis with the baseline models and state-of-the-art system Caliano et al. is also provided.

Table A2: Dev2015: P/R/F/STMM

	BL-D		BL-P		Caliano et al		DT		NB		SVM		BN		MLP		MLR		KNN	
	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.
Precision	0.68	0.66	0.76	0.76	0.64	0.63	0.66	0.72	0.65	0.68	0.65	0.63	0.68	0.68	0.62	0.62	0.63	0.63	0.67	0.67
Recall	0.68	0.66	0.76	0.76	0.64	0.63	0.71	0.69	0.70	0.68	0.72	0.70	0.70	0.68	0.72	0.72	0.72	0.70	0.70	0.69
F-Measure	0.68	0.66	0.76	0.76	0.64	0.63	0.67	0.66	0.67	0.67	0.66	0.66	0.68	0.67	0.66	0.66	0.67	0.66	0.67	0.66
STMM	0.37	0.37	0.37	0.36	0.32	0.32	0.35	0.36	0.36	0.37	0.37	0.36	0.36	0.37	0.37	0.37	0.36	0.36	0.37	0.36

Table A3: Dev2016: P/R/F/STMM

	BL-D		BL-P		Caliano et al		DT		NB		SVM		BN		MLP		MLR		KNN	
	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.
Precision	0.73	0.77	0.89	0.89	0.54	0.54	0.86	0.76	0.78	0.70	0.73	0.72	0.78	0.71	0.73	0.76	0.75	0.73	0.72	0.76
Recall	0.73	0.77	0.89	0.89	0.54	0.54	0.81	0.81	0.73	0.73	0.77	0.69	0.73	0.65	0.77	0.81	0.81	0.77	0.77	0.81
F-Measure	0.73	0.77	0.89	0.89	0.54	0.54	0.80	0.77	0.74	0.70	0.73	0.68	0.74	0.64	0.74	0.77	0.77	0.74	0.73	0.77
STMM	0.46	0.41	0.56	0.56	0.30	0.30	0.54	0.46	0.46	0.32	0.44	0.37	0.46	0.32	0.41	0.46	0.46	0.42	0.48	0.50

Table A4: Test2015 (Wo.C.): P/R/F/STMM

	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Precision	0.64	0.68	0.65	0.71	0.70	0.69	0.73	0.69	0.70	0.74
Recall	0.64	0.68	0.65	0.72	0.72	0.71	0.68	0.75	0.72	0.75
F-Measure	0.64	0.68	0.65	0.71	0.71	0.70	0.70	0.72	0.70	0.73
STMM	0.34	0.33	0.34	0.40	0.33	0.36	0.34	0.37	0.36	0.41

Table A5: Test2016 (Wo.C.): P/R/F/STMM

	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Precision	0.46	0.46	0.31	0.38	0.32	0.39	0.32	0.37	0.37	0.66
Recall	0.46	0.46	0.31	0.54	0.44	0.51	0.44	0.53	0.51	0.48
F-Measure	0.46	0.46	0.31	0.44	0.36	0.43	0.36	0.42	0.42	0.42
STMM	0.36	0.35	0.23	0.43	0.35	0.36	0.35	0.41	0.36	0.38

A.3 Class Wise Accuracy Contribution

This section presents the class wise contribution in terms of accuracy of correctly mapping an entity type in the source ontology to the type in the target ontology. As before, the results for the Dev (W.C. and Wo.C.) and Test (Wo.C.) datasets of #Microposts2015 and #Microposts2016 have been reported here while the remaining results have been reported in Section 3.3. A comparative analysis with the baseline models and state-of-the-art system Caliano et al. is also provided.

Table A6: Accuracy Contribution (%): Dev2015

Entity Type	BL-D		BL-P		Caliano et al		DT		NB		SVM		BN		MLP		MLR		KNN		
	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	
Character	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Event	0.00	0.00	7.69	9.62	7.21	7.21	0.00	1.44	2.88	2.88	0.00	0.00	2.88	2.88	0.00	0.00	0.00	0.00	0.00	0.48	1.44
Location	26.44	24.52	26.44	24.52	23.56	23.56	28.37	28.37	26.44	25.96	28.37	27.40	26.44	25.96	28.85	28.85	28.37	27.40	27.40	28.37	28.37
Organization	7.21	6.73	7.21	6.73	5.77	5.77	10.58	8.65	9.13	8.17	10.10	9.62	9.13	8.17	9.13	9.13	10.10	9.13	9.13	8.65	8.65
Person	33.65	33.17	33.65	33.17	24.52	24.52	31.25	30.29	31.25	30.77	32.69	32.21	31.25	30.77	33.17	33.17	32.69	32.69	32.21	29.81	29.81
Product	0.96	0.96	0.96	0.96	0.00	0.00	0.48	0.48	0.00	0.48	0.96	0.96	0.00	0.48	0.96	0.96	0.96	0.96	0.96	0.96	0.48
Thing	0.00	0.48	0.00	0.48	2.40	2.40	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Overall	68.27	65.87	75.96	75.48	63.46	63.46	70.67	69.23	69.71	68.27	72.12	70.19	69.71	68.27	72.12	72.12	72.12	70.19	70.19	68.75	68.75

Table A7: Accuracy Contribution (%): Dev2016

Entity Type	BL-D		BL-P		Caliano et al		DT		NB		SVM		BN		MLP		MLR		KNN		
	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	W.C.	Wo.C.	
Character	0.00	0.00	0.00	0.00	0.00	0.00	3.85	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Event	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Location	23.08	19.23	23.08	23.08	15.38	15.38	23.08	23.08	23.08	23.08	23.08	23.08	23.08	23.08	23.08	23.08	23.08	23.08	23.08	23.08	23.08
Organization	7.69	7.69	11.54	11.54	7.69	7.69	11.54	11.54	11.54	7.69	11.54	7.69	11.54	11.54	7.69	11.54	11.54	7.69	11.54	11.54	11.54
Person	38.46	46.15	50.00	50.00	26.92	26.92	38.46	42.31	34.62	42.31	38.46	34.62	34.62	30.77	42.31	42.31	42.31	42.31	38.46	42.31	42.31
Product	3.85	3.85	3.85	3.85	3.85	3.85	3.85	3.85	3.85	0.00	3.85	3.85	3.85	0.00	3.85	3.85	3.85	3.85	3.85	3.85	3.85
Thing	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Overall	73.08	76.92	88.46	88.46	53.85	53.85	80.77	80.77	73.08	73.08	76.92	69.23	73.08	65.38	76.92	80.77	80.77	76.92	76.92	80.77	80.77

Table A8: Accuracy Contribution (%): Test2015 (Wo.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00	0.00	0.00	0.00	0.09	0.00	0.00	0.00	0.00	0.00
Event	0.00	3.71	2.91	0.18	0.09	0.00	1.32	0.00	0.00	0.35
Location	37.51	37.51	37.51	43.51	43.78	40.69	41.39	43.42	40.69	43.69
Organization	4.41	4.41	4.41	6.97	7.41	8.74	6.00	7.24	8.47	7.77
Person	20.83	20.83	16.95	19.68	20.39	20.30	18.80	22.33	20.92	20.92
Product	1.50	1.50	0.88	1.24	0.26	1.41	0.44	1.68	1.41	1.59
Thing	0.18	0.00	2.12	0.71	0.00	0.00	0.00	0.00	0.00	0.18
Overall	64.43	67.96	64.78	72.29	72.02	71.14	67.96	74.67	71.49	74.49

Table A9: Accuracy Contribution (%): Test2016 (Wo.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.04
Event	0.00	0.00	1.02	1.02	1.02	0.00	1.02	1.02	0.00	1.02
Location	7.14	7.14	8.16	14.29	11.22	12.24	11.22	13.27	11.22	12.24
Organization	3.06	3.06	0.00	4.08	2.04	4.08	2.04	2.04	4.08	4.08
Person	21.43	21.43	15.31	22.45	18.37	21.43	18.37	23.47	22.45	21.43
Product	13.27	13.27	1.02	12.24	10.20	13.27	10.20	13.27	13.27	7.14
Thing	1.02	1.02	5.10	0.00	1.02	0.00	1.02	0.00	0.00	0.00
Overall	45.92	45.92	30.61	54.08	43.88	51.02	43.88	53.06	51.02	47.96

A.4 Class Wise Precision / Recall

Finally, this section presents the class wise precision and recall scores obtained by correctly mapping an entity mention from a type in the source ontology to the desired type in the target ontology. As before, the results for Train (Wo.C.), Dev (W.C. and Wo.C.) and Test (Wo.C.) datasets of #Microposts2015 and #Microposts2016 have been reported here while the remaining results have been reported in Section 3.4. A comparative analysis with the baseline models and state-of-the-art system Caliano et al. is also provided.

Table A10: Class Wise P/R : Train2015 (Wo.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.80/0.38	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.00/0.00	0.56/0.17	0.08/0.03	0.08/ 0.69	0.00/0.00	0.08/ 0.69	0.00/0.00	0.00/0.00	0.22/0.31
Location	0.86/0.81	0.86/0.81	0.79/0.68	0.81/0.90	0.83/0.86	0.83/0.86	0.83/0.86	0.82/0.91	0.84/0.86	0.79/0.90
Organization	0.77/0.47	0.77/0.47	0.79/0.44	0.63/0.70	0.71/0.62	0.60/ 0.71	0.71/0.61	0.66/ 0.71	0.62/ 0.71	0.68/0.69
Person	0.77/0.86	0.80/0.86	0.94/0.70	0.81/0.85	0.84/0.80	0.79/0.85	0.84/0.80	0.79/0.85	0.78/ 0.87	0.82/0.84
Product	0.34/0.31	0.34/0.31	0.28/0.21	0.59/0.28	0.68/0.14	0.41/0.26	0.65/0.14	0.52/ 0.36	0.45/0.27	0.51/0.31
Thing	0.05/0.28	0.08/0.28	0.05/ 0.67	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00

Table A11: Class Wise P/R : Train2016 (Wo.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.57/0.27	0.07/0.03	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.23/ 0.69	0.87 /0.46	0.64/0.59	0.40/0.59	0.00/0.00	0.41/0.59	0.48/0.40	0.74/0.17	0.64/0.60
Location	0.90 /0.79	0.90 /0.79	0.86/0.73	0.85/0.89	0.86/0.88	0.87/0.85	0.86/0.88	0.86/ 0.90	0.87/0.85	0.83/0.90
Organization	0.70/0.44	0.70/0.44	0.72 /0.41	0.66/0.67	0.64/0.60	0.53/0.69	0.64/0.60	0.63/0.60	0.54/0.67	0.68/0.67
Person	0.75/0.86	0.75/0.86	0.92 /0.69	0.80/0.86	0.82/0.79	0.75/0.84	0.82/0.79	0.76/ 0.88	0.75/0.86	0.79/0.84
Product	0.36/ 0.34	0.36/ 0.34	0.26/0.21	0.47/0.33	0.27/0.25	0.42/0.30	0.27/0.25	0.43/0.29	0.45/0.31	0.58/0.34
Thing	0.03/0.22	0.03/0.04	0.06/ 0.77	0.17/0.06	0.02/0.03	0.00/0.00	0.03/0.04	0.33/0.01	0.00/0.00	0.39 /0.13

Table A12: Class Wise P/R : Dev2015 (W.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.64/ 0.64	1.00 /0.60	0.00/0.00	0.30/0.24	0.00/0.00	0.30/0.24	0.00/0.00	0.00/0.00	0.33/0.04
Location	0.92 /0.82	0.92 /0.82	0.82/0.73	0.89/0.88	0.73/0.82	0.88/0.88	0.89/0.82	0.85/ 0.90	0.88/0.88	0.83/0.85
Organization	0.68 /0.56	0.68 /0.56	0.60/0.44	0.34/ 0.81	0.51/0.70	0.41/0.78	0.38/0.70	0.48/0.70	0.47/0.78	0.39/0.70
Person	0.78/ 0.95	0.78/ 0.95	0.81/0.69	0.88 /0.88	0.87/0.88	0.82/0.92	0.87/0.88	0.76/0.93	0.78/0.92	0.84/0.91
Product	0.33/ 0.25	0.33/ 0.25	0.00/0.00	0.33/0.13	0.00/0.00	0.29/ 0.25	0.00/0.00	0.40/0.25	0.22/0.25	0.40/0.25
Thing	0.00/0.00	0.00/0.00	0.12/1.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00

Table A13: Class Wise P/R : Dev2015 (Wo.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.61/ 0.80	1.00 /0.60	1.00 /0.12	0.30/0.24	0.00/0.00	0.30/0.24	0.00/0.00	0.00/0.00	0.50/0.12
Location	0.88 /0.76	0.88 /0.76	0.82/0.73	0.78/0.88	0.84/0.81	0.85/0.85	0.84/0.81	0.85/ 0.90	0.85/0.85	0.80/0.88
Organization	0.67 /0.52	0.67 /0.52	0.60/0.44	0.38/0.67	0.37/0.63	0.36/ 0.74	0.37/0.63	0.43/0.70	0.36/0.70	0.42/0.67
Person	0.82/ 0.93	0.82/ 0.93	0.81/0.69	0.83/0.85	0.84 /0.86	0.83/0.91	0.84 /0.86	0.78/ 0.93	0.82/0.92	0.81/0.84
Product	0.29/ 0.25	0.29/ 0.25	0.00/0.00	0.20/0.13	0.50 /0.13	0.40/ 0.25	0.50 /0.13	0.40/ 0.25	0.40/ 0.25	0.20/0.13
Thing	0.03/0.20	0.20 /0.20	0.12/ 1.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00

Table A14: Class Wise P/R : Dev2016 (W.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.00/0.00	1.00/0.33	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Location	0.86/ 1.00	1.00/1.00	1.00 /0.67	0.67/ 1.00	0.86/ 1.00	0.86/ 1.00	0.86/ 1.00	0.86/ 1.00	0.86/ 1.00	0.67/ 1.00
Organization	1.00 /0.67	1.00/1.00	0.50/0.67	1.00/1.00	1.00/1.00	0.50/ 1.00	1.00/1.00	0.50/0.67	0.60/ 1.00	0.60/ 1.00
Person	0.83/0.77	0.81/ 1.00	1.00 /0.54	0.91/0.77	0.90/0.69	0.91/0.77	0.90/0.69	0.92/0.85	0.92/0.85	0.91/0.77
Product	0.50/ 1.00	1.00/1.00	0.33/ 1.00	0.50/ 1.00	0.33/ 1.00	0.50/ 1.00	0.33/ 1.00	0.33/ 1.00	0.50/ 1.00	1.00/1.00
Thing	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00

Table A15: Class Wise P/R : Dev2016 (Wo.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Location	0.83/0.83	1.00/1.00	1.00/0.67	0.75/ 1.00	0.60/ 1.00	0.75/ 1.00	0.67/ 1.00	0.75/ 1.00	0.67/ 1.00	0.67/ 1.00
Organization	0.67/0.67	1.00/1.00	0.50/0.67	0.60/ 1.00	0.50/0.67	0.33/0.67	0.50/ 1.00	0.60/ 1.00	0.50/0.67	0.60/ 1.00
Person	1.00/0.92	0.81/ 1.00	1.00/0.50	1.00/0.85	1.00/0.85	1.00/0.69	1.00/0.62	1.00/0.85	1.00/0.85	1.00/0.85
Product	0.25/ 1.00	1.00/1.00	0.33/ 1.00	0.50/ 1.00	0.00/0.00	0.33/ 1.00	0.00/0.00	0.50/ 1.00	0.50/ 1.00	1.00/1.00
Thing	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00

Table A16: Class Wise P/R : Test2015 (Wo.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.04/0.14	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00
Event	0.00/0.00	0.33/ 0.69	0.89/0.54	0.11/0.03	0.14/0.02	0.00/0.00	0.08/0.25	0.00/0.00	0.00/0.00	0.17/0.07
Location	0.94/0.77	0.94/0.77	0.92/0.77	0.89/0.89	0.91/ 0.90	0.92/0.83	0.92/0.85	0.89/0.89	0.93/0.83	0.91/ 0.90
Organization	0.51/0.32	0.51/0.32	0.55/0.32	0.41/0.51	0.43/0.54	0.38/ 0.63	0.41/0.44	0.47/0.53	0.38/0.62	0.45/0.56
Person	0.67/0.83	0.67/0.83	0.90/0.67	0.75/0.78	0.74/0.81	0.69/0.80	0.78/0.74	0.70/ 0.88	0.68/0.83	0.75/0.83
Product	0.31/0.40	0.31/0.40	0.26/0.23	0.27/0.33	0.08/0.07	0.43/0.37	0.56/0.12	0.44/ 0.44	0.47/0.37	0.55/0.42
Thing	0.01/0.07	0.00/0.00	0.08/ 0.89	0.44/0.30	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.50/0.07

Table A17: Class Wise P/R : Test2016 (Wo.C.)

Entity Type	BL-D	BL-P	Caliano et al	DT	NB	SVM	BN	MLP	MLR	KNN
Character	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	0.00/0.00	1.00/0.07
Event	0.00/0.00	0.00/0.00	0.08/ 0.33	0.33/ 0.33	0.25/ 0.33	0.00/0.00	0.25/ 0.33	0.50/0.33	0.00/0.00	0.33/ 0.33
Location	0.78/0.47	0.78/0.47	0.73/0.53	0.70/ 0.93	0.61/0.73	0.86/0.80	0.61/0.73	0.72/0.87	0.79/0.73	0.60/0.80
Organization	0.43/0.43	0.43/0.43	0.00/0.00	0.33/ 0.57	0.20/0.29	0.14/ 0.57	0.20/0.29	0.22/0.29	0.17/ 0.57	0.17/ 0.57
Person	0.49/0.91	0.49/0.91	0.83/0.65	0.46/0.96	0.58/0.78	0.55/0.91	0.58/0.78	0.43/ 1.00	0.52/0.96	0.51/0.91
Product	0.68/ 0.87	0.68/ 0.87	0.13/0.07	0.86/0.80	0.33/0.67	0.76/ 0.87	0.33/0.67	0.81/ 0.87	0.76/ 0.87	0.88/0.47
Thing	0.05/0.17	0.05/0.17	0.11/ 0.83	0.00/0.00	0.20/0.17	0.00/0.00	0.20/0.17	0.00/0.00	0.00/0.00	0.00/0.00

B Learning2Link

This section presents the results for the Learning2Link approach that has been proposed in Chapter 3 Section 3.4 with the intent to improve the performance of a linking component, thereby improving the overall performance of a Named Entity Extraction and Linking framework, using evidence from the recognition and disambiguation components of the given framework. The results are reported in terms of F-Measure and SLM (i.e., Strong Link Match)

for when *ten* candidate resources have been considered for an entity mention in #Microposts2015 and #Microposts2016 datasets. Learning2Link has been performed for entity mentions that have either been identified using a named entity recognition system (such as T-NER) or obtained by using a NER Oracle.

Table B1: #Microposts2015: F-Measure (10 instances)

Learning Models	Train 2015		Dev 2015		Test 2015	
	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)
BN	0.68	0.98	0.69	0.98	0.73	0.98
NB	0.55	0.97	0.53	0.97	0.64	0.98
SVM	0.51	0.98	0.44	0.98	0.64	0.98
MLP	0.66	0.98	0.44	0.98	0.78	0.99
VP	0.55	0.98	0.44	0.98	0.66	0.98
KNN	0.73	0.98	0.68	0.98	0.70	0.98
DT	0.80	0.99	0.67	0.98	0.69	0.98

Table B2: #Microposts2016: F-Measure (10 instances)

Learning Models	Train 2016		Dev 2016		Test 2016	
	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)
BN	0.71	0.98	0.53	0.98	0.21	0.96
NB	0.57	0.97	0.35	0.97	0.20	0.96
SVM	0.57	0.98	0.32	0.98	0.21	0.98
MLP	0.71	0.99	0.65	0.99	0.25	0.97
VP	0.59	0.98	0.39	0.98	0.22	0.98
KNN	0.77	0.99	0.59	0.98	0.25	0.97
DT	0.82	0.99	0.36	0.98	0.26	0.97

Table B3: #Microposts2015: F-Measure (10 instances) - With Oracle

Learning Models	Train 2015		Dev 2015		Test 2015	
	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)
BN	0.68	0.98	0.63	0.98	0.68	0.98
NB	0.55	0.96	0.48	0.97	0.60	0.97
SVM	0.55	0.97	0.45	0.98	0.63	0.98
MLP	0.65	0.98	0.66	0.98	0.70	0.99
VP	0.55	0.97	0.46	0.98	0.63	0.98
KNN	0.78	0.98	0.63	0.98	0.69	0.98
DT	0.81	0.99	0.73	0.99	0.70	0.98

Table B4: #Microposts2016: F-Measure (10 instances) - With Oracle

Learning Models	Train 2016		Dev 2016		Test 2016	
	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)	F ₁ (Yes)	F ₁ (No)
BN	0.68	0.98	0.54	0.97	0.21	0.96
NB	0.57	0.97	0.32	0.96	0.20	0.96
SVM	0.58	0.98	0.34	0.97	0.21	0.97
MLP	0.67	0.98	0.36	0.97	0.23	0.97
VP	0.60	0.98	0.35	0.97	0.23	0.97
KNN	0.79	0.99	0.50	0.97	0.23	0.96
DT	0.82	0.99	0.57	0.98	0.28	0.97

Table B5: Performance Analysis: L2L (T-NER) vs L2L (Oracle) - 10 instances

Datasets	#Microposts2015		#Microposts2016	
	L2L (T-NER)	L2L (Oracle)	L2L (T-NER)	L2L (Oracle)
Train	0.89	0.90	0.90	0.90
Dev	0.83	0.86	0.82	0.77
Test	0.88	0.84	0.62	0.62

Table B6: #Microposts2015: SLM (10 instances)

Learning Models	Train 2015			Dev 2015			Test 2015		
	SLM1	SLM2	NIL	SLM1	SLM2	NIL	SLM1	SLM2	NIL
BN	0.28	0.74	0.65	0.23	0.72	0.71	0.28	0.74	0.72
NB	0.23	0.62	0.65	0.18	0.55	0.77	0.23	0.62	0.72
SVM	0.15	0.39	0.87	0.10	0.30	0.93	0.15	0.39	0.94
MLP	0.22	0.60	0.82	0.10	0.30	0.86	0.22	0.60	0.87
VP	0.17	0.45	0.77	0.10	0.32	0.90	0.17	0.45	0.92
KNN	0.27	0.72	0.77	0.21	0.64	0.84	0.27	0.72	0.80
DT	0.29	0.79	0.83	0.19	0.60	0.82	0.29	0.79	0.84

Table B7: #Microposts2016: SLM (10 instances)

Learning Models	Train 2016			Dev 2016			Test 2016		
	SLM1	SLM2	NIL	SLM1	SLM2	NIL	SLM1	SLM2	NIL
BN	0.32	0.99	0.73	0.12	0.62	0.75	0.05	0.26	0.64
NB	0.28	0.65	0.72	0.08	0.38	0.66	0.04	0.22	0.60
SVM	0.19	0.44	0.92	0.04	0.22	0.65	0.03	0.15	0.83
MLP	0.28	0.65	0.87	0.14	0.68	0.82	0.05	0.25	0.69
VP	0.21	0.48	0.89	0.06	0.28	0.69	0.03	0.16	0.80
KNN	0.34	0.78	0.83	0.12	0.62	0.78	0.05	0.27	0.67
DT	0.34	0.79	0.88	0.06	0.28	0.67	0.04	0.21	0.74

Table B8: #Microposts2015: SLM (10 instances) - With Oracle

Learning Models	Train 2015		Dev 2015		Test 2015	
	SLM	NIL	SLM	NIL	SLM	NIL
BN	0.69	0.68	0.68	0.72	0.71	0.74
NB	0.57	0.65	0.53	0.75	0.65	0.77
SVM	0.41	0.89	0.32	0.92	0.50	0.93
MLP	0.59	0.80	0.57	0.90	0.59	0.93
VP	0.43	0.87	0.34	0.91	0.50	0.93
KNN	0.77	0.79	0.63	0.76	0.68	0.78
DT	0.78	0.84	0.73	0.82	0.67	0.82

Table B9: #Microposts2016: SLM (10 instances) - With Oracle

Learning Models	Train 2016		Dev 2016		Test 2016	
	SLM	NIL	SLM	NIL	SLM	NIL
BN	0.68	0.68	0.51	0.51	0.27	0.27
NB	0.62	0.62	0.27	0.27	0.24	0.24
SVM	0.45	0.45	0.21	0.21	0.15	0.15
MLP	0.58	0.58	0.24	0.24	0.21	0.21
VP	0.48	0.48	0.22	0.22	0.18	0.18
KNN	0.78	0.78	0.49	0.49	0.27	0.27
DT	0.77	0.77	0.45	0.45	0.27	0.27

B.1 Learning2Link for Italian Language Tweets

As an additional experiment, Learning2Link has been used to evaluate the linking performance of entity mentions recognized from Italian language tweets. The training dataset consists of 1000 tweets made available by the EVALITA 2016 NEEL-IT challenge [9]. CRF has been used to identify entity mentions from the given dataset and classified using the ontology O_T . In particular, two configurations of CRF have been trained using the training data available for the challenge: (1) CRF and (2) CRF+Gazetteers. In the second configuration, the model has been induced enclosing several gazetteers, i.e. products, organizations, persons, events and characters. The output of CRF is a set of entity mentions e_1, e_2, \dots, e_m in a given tweet t .

CRF identifies a total of 313 entity mentions from the training dataset (as opposed to 800 mentions in the ground truth provided by the challenge organizers). These mentions are then used to create an input space to train the models so as to predict the *target class* for mentions identified from the test set. 803 entity mentions have been identified from the test set (of 1500 tweets) using CRF. Further, these mentions are linked using Learning2Link, for which Italian version of DBpedia has been indexed to create a local index similar to how it has been created for the English tweets. A 10-fold cross validation is performed where the models learn if an entity mention is linkable or unlinkable, and based on the *Decision Criteria* used in Section 3.2,

the most suitable candidate resource for the entity mention from the KB is predicted or the entity mention is marked as a NIL mention if no candidate matches are found.

The results of entity recognition by CRF, in terms of Precision (P), Recall (R) and F1-Measure (F1) have been reported in Table B10, according to two investigated configurations: CRF and CRF+Gazetteers. A first observation is the poor recognition performances obtained in both configurations, which are mainly due to the limited amount of entity mentions in the training data. These poor performances are highlighted even more by looking at the entity types *Thing* (20), *Event* (15) and *Character* (18), whose limited number of instances do not allow CRF to learn any linguistic pattern to recognize them. For the remaining types, CRF+Gazetteers is able to improve Precision but at the expense of Recall.

Table B10: Entity Recognition

Entity Type	CRF			CRF+Gazetteers		
	Precision	Recall	F-Measure	Precision	Recall	F-Measure
Character	0.00	0.00	0.00	0.00	0.00	0.00
Event	0.00	0.00	0.00	0.00	0.00	0.00
Location	0.56	0.40	0.47	0.64	0.40	0.50
Organization	0.43	0.24	0.31	0.60	0.20	0.30
Person	0.50	0.30	0.37	0.69	0.21	0.33
Product	0.12	0.11	0.11	0.31	0.10	0.16
Thing	0.00	0.00	0.00	0.00	0.00	0.00
Overall	0.37	0.24	0.29	0.57	0.20	0.30

The low recognition performance have a great impact on the subsequent linking part as well. Therefore, the results of entity linking by Learning2Link have been reported by considering a NER oracle (i.e. a perfect named entity recognition system). The Precision (P), Recall (R) and F-measure (F1) for the Strong Link Match (SLM) measure for Learning2Link have been reported in Table B11. Although the performances are low in terms of F-measure, it can be observed that Decision Tree (DT) is a leaner algorithm with the highest Strong Link Match F-measure. On the other hand, low recall scores could

be attributed to the inability of the retrieval system to find the “correct” link in the top-10 candidate list. A list of irrelevant candidate resources results in uninformative similarity scores, which causes the learning models to predict a target class where none of the candidate resources is a suitable match for an entity mention.

Table B11: Strong Link Match measure

Learning Models	Precision	Recall	F-Measure
DT	0.733	0.371	0.492
MLP	0.684	0.333	0.448
NB	0.614	0.312	0.414
MLR	0.709	0.278	0.399
SVM-Poly.	0.721	0.270	0.393
VP	0.696	0.274	0.393
BN	0.741	0.266	0.392
SVM-Radial	0.724	0.264	0.387
SVM-Linear	0.686	0.266	0.384

ACKNOWLEDGEMENTS

*“Some people grumble that roses have thorns;
I am grateful that thorns have roses.”*

- Alphonse Karr

Pursuing research is not just about writing a few pieces of code, and putting together some results in scientific papers. It is so much more than that. Over the last three years, I have learned how intellectually engaging and rewarding a research career can be, when you are in safe hands. Having said that, I would like to extend my highest gratitude towards my respected and hard-working supervisors, Dr. Elisabetta Fersini and Dr. Matteo Palmonari. Without their guidance, be it day or night, I wouldn't have been able to succeed in pursuing my research period in this university. Over time, I have constantly badgered them with questions and dutifully, politely and patiently, they have taught me the whats, whys and hows of doing things with perfection and perseverance.

I would also like to acknowledge the support and guidance of Prof. Enza Messina, who has been there for us students of the MIND Lab as a constant pillar of support and strength. Further, I would like to thank my tutor Prof. Giancarlo Mauri, who has been very patient since the very starting of my research career in this university and has guided me through different stages, every step of the way. I am also grateful to the research staff and our highly respected PhD Coordinator at our Department of Informatics, Systems and Communication, Prof. Stefania Bandini, who have been very cooperative to all the PhD students and provided an unbiased and productive environment for a healthy research.

I can not forget to mention my colleagues in this department, viz, Debora Nozza, Flavio Cecchini, Augusto Ballardini, Blerina Spahiu and Simone Fontana who have been friendly, kind and helpful to me whenever I needed. I want to specifically express my love and gratitude for my friend and senior researcher at the department Dr. Görkem Kılınç who's calming and gentle

presence has helped me through the ups and downs of my research period in these past years. Her support and friendship means the world to me, here's hoping for many more years of it!

Finally, and most importantly, I would like to thank my mother, my father, my little brother Gauraang and my friends (Palak, Sunny, Abhishek, Nisha, Poulomi, Jyoti, Arpita, and Sudharshan) who have sat with me patiently as I gave my blood, sweat and tears over the years to achieve fruitful results. Although oceans apart, I have always felt their warm love and affection in my heart. They have constantly been my backbone and their kind, wise words have inspired me to keep going and have faith in myself even in the face of failure. Their trust in me has always kept me grounded and I thank the Lord for blessing me with such a loving and caring family.

All my accomplishments and success would not have been possible if I weren't the Chosen One by the Lord, Almighty, for all we humans are nothing but actors in the stage of life, so beautifully designed by our Creator. Not a day goes by when I am not grateful to God for blessing me with opportunities to realize my full potential in this journey of life!

Praise the Lord!

Bibliography

- [1] ARNOLD, A., NALLAPATI, R., AND COHEN, W. W. Exploiting feature hierarchy for transfer learning in named entity recognition. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, (ACL), Ohio, USA* (2008), pp. 245–253.
- [2] ATENCIA, M., BORGIDA, A., EUZENAT, J., GHIDINI, C., AND SERAFINI, L. A formal semantics for weighted ontology mappings. In *11th International Semantic Web Conference (ISWC), Boston, USA* (2012), Springer, pp. 17–33.
- [3] ATZMUELLER, M., KLUEGL, P., AND PUPPE, F. Rule-based information extraction for structured data acquisition using textmarker. In *LWA - Workshop-Woche: Lernen, Wissen & Adaptivität, Würzburg* (2008), pp. 1–7.
- [4] AUER, S., BIZER, C., KOBILAROV, G., LEHMANN, J., CYGANIAK, R., AND IVES, Z. Dbpedia: A nucleus for a web of open data. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007, Busan, Korea*. Springer, 2007, pp. 722–735.
- [5] AUGENSTEIN, I., MAYNARD, D., AND CIRAVEGNA, F. Relation extraction from the web using distant supervision. In *Knowledge Engineering and Knowledge Management - 19th International Conference, (EKAW), Linköping, Sweden*. Springer, 2014, pp. 26–41.
- [6] BAADER, F., AND NUTT, W. Basic description logics. In *The Description Logic Handbook: Theory, Implementation, and Applications* (2003), pp. 43–95.
- [7] BALDWIN, T., HAN, B., DE MARNEFFE, M., KIM, Y., RITTER, A., AND XU, W. Findings of the 2015 workshop on noisy user-generated

- text. In *Proceedings of the Workshop on Noisy User-generated Text (WNUT)*. Association for Computational Linguistics (2015).
- [8] BASAVE, A. C., RIZZO, G., VARGA, A., ROWE, M., STANKOVIC, M., AND DADZIE, A.-S. Making sense of microposts (# microposts2014) named entity extraction & linking challenge. In *Proceedings of the 4th Workshop on Making Sense of Microposts (# Microposts2014) co-located with the 23rd International World Wide Web Conference (WWW)*, Seoul, Korea (2014), pp. 54–60.
- [9] BASILE, P., CAPUTO, A., GENTILE, A. L., AND RIZZO, G. Overview of the EVALITA 2016 named entity recognition and linking in italian tweets (NEEL-IT) task.
- [10] BASILE, P., CAPUTO, A., AND SEMERARO, G. An enhanced lesk word sense disambiguation algorithm through a distributional semantic model. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING)*, Dublin, Ireland (2014), pp. 1591–1600.
- [11] BASILE, P., CAPUTO, A., SEMERARO, G., AND NARDUCCI, F. UNIBA: exploiting a distributional semantic model for disambiguating and linking entities in tweets. In *Proceedings of the the 5th Workshop on Making Sense of Microposts co-located with the 24th International World Wide Web Conference (WWW)*, Florence, Italy (2015), p. 62.
- [12] BENAJIBA, Y., ROSSO, P., AND BENEDÍRUIZ, J. M. Anersys: An arabic named entity recognition system based on maximum entropy. In *8th International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City (2007), Springer, pp. 143–153.
- [13] BIZER, C., HEATH, T., AND BERNERS-LEE, T. Linked data—the story so far. *International Journal on Semantic Web and Information Systems* 5, 3 (2009), 1–22.
- [14] BONTCHEVA, K., DERCZYNSKI, L., FUNK, A., GREENWOOD, M. A., MAYNARD, D., AND ASWANI, N. Twitie: An open-source information extraction pipeline for microblog text. In *Recent Advances*

- in Natural Language Processing (RANLP), Hissar, Bulgaria* (2013), pp. 83–90.
- [15] BONTCHEVA, K., TABLAN, V., MAYNARD, D., AND CUNNINGHAM, H. Evolving gate to meet new challenges in language engineering. *Natural Language Engineering* 10, 3-4 (2004), 349–373.
- [16] BREIMAN, L., FRIEDMAN, J., OLSHEN, R. A., AND STONE, C. J. *Classification and Regression Trees*. CRC press, 1984.
- [17] BUNDSCHUS, M., DEJORI, M., STETTER, M., TRESP, V., AND KRIEGEL, H.-P. Extraction of semantic biomedical relations from text using conditional random fields. *BMC bioinformatics* 9, 1 (2008), 1.
- [18] CALIANO, D., FERSINI, E., MANCHANDA, P., PALMONARI, M., AND MESSINA, E. Unimib: Entity linking in tweets using jaro-winkler distance, popularity and coherence. 70–72.
- [19] CANO BASAVE, A. E., VARGA, A., ROWE, M., STANKOVIC, M., AND DADZIE, A.-S. Making sense of microposts (# msm2013) concept extraction challenge. 1–15.
- [20] CHANG, M.-W., HSU, B.-J., MA, H., LOYND, R., AND WANG, K. E2e: An end-to-end entity linking system for short and noisy text. 62–63.
- [21] CHERRY, C., AND GUO, H. The unreasonable effectiveness of word representations for twitter named entity recognition. In *NAACL HLT, The 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Colorado, USA* (2015), pp. 735–745.
- [22] CIMIANO, P., AND VÖLKER, J. Towards large-scale, open-domain and ontology-based named entity classification. In *In Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP), Borovets, Bulgaria* (2005), vol. 5, pp. 66–166.

- [23] COHEN, W., RAVIKUMAR, P., AND FIENBERG, S. A comparison of string metrics for matching names and records. In *KDD workshop on data cleaning and object consolidation* (2003), vol. 3, pp. 73–78.
- [24] CORTES, C., AND VAPNIK, V. Support-vector networks. *Machine learning* (1995), 273–297.
- [25] CUCERZAN, S. Large-scale named entity disambiguation based on wikipedia data. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning EMNLP-CoNLL, Prague, Czech Republic* (2007), vol. 7, pp. 708–716.
- [26] CUNNINGHAM, H., MAYNARD, D., BONTCHEVA, K., AND TABLAN, V. A framework and graphical development environment for robust nlp tools and applications. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Philadelphia, USA.* (2002), pp. 168–175.
- [27] CUNNINGHAM, H., MAYNARD, D., BONTCHEVA, K., AND TABLAN, V. Gate: an architecture for development of robust hlt applications. In *Proceedings of the 40th annual meeting on Association for Computational Linguistics (ACL)* (2002), pp. 168–175.
- [28] DAMLJANOVIC, D., AND BONTCHEVA, K. Named entity disambiguation using linked data. In *Proceedings of the 9th Extended Semantic Web Conference ESWC, Heraklion, Greece* (2012).
- [29] DAUMÉ III, H. Frustratingly easy domain adaptation. *CoRR abs/0907.1815* (2009).
- [30] DERCZYNSKI, L., AUGENSTEIN, I., AND BONTCHEVA, K. Usfd: Twitter ner with drift compensation and linked data. *CoRR abs/1511.03088* (2015).
- [31] DERCZYNSKI, L., MAYNARD, D., RIZZO, G., VAN ERP, M., GORRELL, G., TRONCY, R., PETRAK, J., AND BONTCHEVA, K. Analysis

- of named entity recognition and linking for tweets. *Information Processing & Management* 51, 2 (2015), 32–49.
- [32] DOAN, A., MADHAVAN, J., DOMINGOS, P., AND HALEVY, A. Learning to map between ontologies on the semantic web. In *Proceedings of the 11th international conference on World Wide Web (WWW), Honolulu, Hawaii* (2002), ACM, pp. 662–673.
- [33] DUDA, R. O., HART, P. E., AND STORK, D. G. Pattern classification, 2nd edition. *J. Classification* 18, 2 (2001), 273–275.
- [34] EHRIG, M., AND STAAB, S. Qom–quick ontology mapping. In *Proceedings of the 3rd International Semantic Web Conference, (ISWC), Hiroshima, Japan* (2004), Springer, pp. 683–697.
- [35] EHRIG, M., AND SURE, Y. Ontology mapping—an integrated approach. In *The Semantic Web: Research and Applications, First European Semantic Web Symposium (ESWS), Heraklion, Greece* (2004), Springer, pp. 76–91.
- [36] EK, T., KIRKEGAARD, C., JONSSON, H., AND NUGUES, P. Named entity recognition for short text messages. *Procedia-Social and Behavioral Sciences* 27 (2011), 178–187.
- [37] ETZIONI, O., CAFARELLA, M., DOWNEY, D., POPESCU, A.-M., SHAKED, T., SODERLAND, S., WELD, D. S., AND YATES, A. Un-supervised named-entity extraction from the web: An experimental study. *Artificial intelligence* 165, 1 (2005), 91–134.
- [38] FANG, Y., AND CHANG, M.-W. Entity linking on microblogs with spatial and temporal signals. *Transactions of the Association for Computational Linguistics* 2 (2014), 259–272.
- [39] FELLBAUM, C. *WordNet*. Wiley Online Library, 1998.
- [40] FERRAGINA, P., AND SCAIELLA, U. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management CIKM, Toronto, Canada* (2010), pp. 1625–1628.

- [41] FEYISETAN, O., LUCZAK-ROESCH, M., SIMPERL, E., TINATI, R., AND SHADBOLT, N. Towards hybrid ner: a study of content and crowdsourcing-related performance factors. In *12th European Semantic Web Conference ESWC, Portoroz, Slovenia* (2015), pp. 525–540.
- [42] FININ, T., MURNANE, W., KARANDIKAR, A., KELLER, N., MARTINEAU, J., AND DREDZE, M. Annotating named entities in twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk* (2010), pp. 80–88.
- [43] FINKEL, J. R., GRENAGER, T., AND MANNING, C. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics (ACL), University of Michigan, USA* (2005), Association for Computational Linguistics, pp. 363–370.
- [44] FRANCIS-LANDAU, M., DURRETT, G., AND KLEIN, D. Capturing semantic similarity for entity linking with convolutional neural networks. *CoRR abs/1604.00734* (2016).
- [45] FREUND, Y., AND SCHAPIRE, R. E. Large margin classification using the perceptron algorithm. *Machine learning* 37, 3 (1999), 277–296.
- [46] FROMREIDE, H., HOVY, D., AND SØGAARD, A. Crowdsourcing and annotating ner for twitter# drift. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation LREC, Reykjavik, Iceland* (2014), pp. 2544–2547.
- [47] GATTANI, A., LAMBA, D. S., GARERA, N., TIWARI, M., CHAI, X., DAS, S., SUBRAMANIAM, S., RAJARAMAN, A., HARINARAYAN, V., AND DOAN, A. Entity extraction, linking, classification, and tagging for social media: a wikipedia-based approach. *Proceedings of the VLDB Endowment* 6, 11 (2013), 1126–1137.
- [48] GHAHRAMANI, Z., AND JORDAN, M. I. Factorial hidden markov models. *Machine learning* 29, 2-3 (1997), 245–273.

- [49] GIARETTA, P., AND GUARINO, N. Ontologies and knowledge bases towards a terminological clarification. *Towards very large knowledge bases: knowledge building & knowledge sharing 25* (1995), 32.
- [50] GORRELL, G., PETRAK, J., AND BONTCHEVA, K. Using@ twitter conventions to improve #lod-based named entity disambiguation. In *The Semantic Web. Latest Advances and New Domains. Proceedings of the 12th European Semantic Web Conference (ESWC), Portoroz, Slovenia*. Springer, 2015, pp. 171–186.
- [51] GRISHMAN, R., AND SUNDHEIM, B. Message understanding conference-6: A brief history. In *Proceedings of the 16th conference on Computational linguistics - Volume 1 (COLING)* (1996), vol. 96, pp. 466–471.
- [52] GUARINO, N. Formal ontology and information systems. In *Proceedings of the 1st International Conference, Trento, Italy* (1998), vol. 98, pp. 81–97.
- [53] GUO, H., ZHU, H., GUO, Z., ZHANG, X., WU, X., AND SU, Z. Domain adaptation with latent semantic association for named entity recognition. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Colorado, USA* (2009), pp. 281–289.
- [54] GUO, S., CHANG, M.-W., AND KICIMAN, E. To link or not to link? a study on end-to-end tweet entity linking. In *Human Language Technologies: Conference of the North American Chapter of the Association of Computational Linguistics, Proceedings, Atlanta, USA* (2013), pp. 1020–1030.
- [55] HABIB, M. B., AND KEULEN, M. Unsupervised improvement of named entity extraction in short informal context using disambiguation clues. 1–10.
- [56] HABIB, M. B., AND VAN KEULEN, M. Need4tweet: A twitterbot for tweets named entity extraction and disambiguation. 31–36.

- [57] HACHEY, B., RADFORD, W., NOTHMAN, J., HONNIBAL, M., AND CURRAN, J. R. Evaluating entity linking with wikipedia. *Artificial intelligence 194* (2013), 130–150.
- [58] HANISCH, D., FUNDEL, K., MEVISSSEN, H.-T., ZIMMER, R., AND FLUCK, J. Prominer: rule-based protein and gene entity recognition. *BMC bioinformatics 6*, Suppl 1 (2005), S14.
- [59] HASEGAWA, T., SEKINE, S., AND GRISHMAN, R. Discovering relations among named entities from large corpora. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain* (2004), Association for Computational Linguistics, p. 415.
- [60] HOFFART, J., SEUFERT, S., NGUYEN, D. B., THEOBALD, M., AND WEIKUM, G. Kore: keyphrase overlap relatedness for entity disambiguation. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, USA* (2012), ACM, pp. 545–554.
- [61] HOFFART, J., YOSEF, M. A., BORDINO, I., FÜRSTENAU, H., PINKAL, M., SPANIOL, M., TANEVA, B., THATER, S., AND WEIKUM, G. Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing EMNLP, Edinburgh, UK* (2011), Association for Computational Linguistics, pp. 782–792.
- [62] HUA, W., ZHENG, K., AND ZHOU, X. Microblog entity linking with social temporal context. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Australia* (2015), ACM, pp. 1761–1775.
- [63] HUANG, H., HECK, L., AND JI, H. Leveraging deep neural networks and knowledge graphs for entity disambiguation. *CoRR abs/1504.07678* (2015).
- [64] JARO, M. A. Probabilistic linkage of large public health data files. *Statistics in medicine 14*, 5-7 (1995), 491–498.

- [65] KARATAY, D., AND KARAGOZ, P. User interest modeling in twitter with named entity recognition. 17–20.
- [66] KOULOUMPIS, E., WILSON, T., AND MOORE, J. D. Twitter sentiment analysis: The good the bad and the omg! 538–541.
- [67] KOZAREVA, Z. Bootstrapping named entity recognition with automatically generated gazetteer lists. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics EACL, Trento, Italy* (2006), Association for Computational Linguistics, pp. 15–21.
- [68] KULKARNI, S., SINGH, A., RAMAKRISHNAN, G., AND CHAKRABARTI, S. Collective annotation of wikipedia entities in web text. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France* (2009), ACM, pp. 457–466.
- [69] LAFFERTY, J. D., MCCALLUM, A., AND PEREIRA, F. C. N. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning (ICML), Williamstown, USA* (2001), pp. 282–289.
- [70] LANDWEHR, N., HALL, M., AND FRANK, E. Logistic model trees. *Machine learning* (2005), 161–205.
- [71] LI, C., WENG, J., HE, Q., YAO, Y., DATTA, A., SUN, A., AND LEE, B.-S. Twiner: named entity recognition in targeted twitter stream. In *The 35th International ACM SIGIR conference on research and development in Information Retrieval, Portland, USA* (2012), pp. 721–730.
- [72] LI, W., AND MCCALLUM, A. Rapid development of hindi named entity recognition using conditional random fields and feature induction. *ACM Transactions on Asian Language Information Processing (TALIP)* 2, 3 (2003), 290–294.

- [73] LIU, X., LI, Y., WU, H., ZHOU, M., WEI, F., AND LU, Y. Entity linking for tweets. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, (ACL), Bulgaria, Volume 1* (2013), pp. 1304–1311.
- [74] LIU, X., ZHANG, S., WEI, F., AND ZHOU, M. Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, (ACL), Portland, USA* (2011), pp. 359–367.
- [75] LOCKE, B. W. Named entity recognition: Adapting to microblogging.
- [76] MAYNARD, D., TABLAN, V., URSU, C., CUNNINGHAM, H., AND WILKS, Y. Named entity recognition from diverse text types. In *In Recent Advances in Natural Language Processing 2001 Conference, Tzigov Chark* (2001), pp. 257–274.
- [77] MCCALLUM, A. A comparison of event models for naive bayes text classification. In *AAAI-98 Workshop on Learning for Text Categorization* (1998).
- [78] MCCALLUM, A., FREITAG, D., AND PEREIRA, F. C. Maximum entropy markov models for information extraction and segmentation. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML), Stanford University, USA* (2000), vol. 17, pp. 591–598.
- [79] MCCALLUM, A., AND LI, W. 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, CoNLL, Held in cooperation with HLT-NAACL, Edmonton, Canada* (2003), Association for Computational Linguistics, pp. 188–191.
- [80] MEIJ, E., WEERKAMP, W., AND DE RIJKE, M. Adding semantics to microblog posts. In *Proceedings of the Fifth International Conference on Web Search and Data Mining WSDM, Seattle, USA* (2012), pp. 563–572.

- [81] MILLER, G. A., BECKWITH, R., FELLBAUM, C., GROSS, D., AND MILLER, K. J. Introduction to wordnet: An on-line lexical database. *International journal of lexicography* 3, 4 (1990), 235–244.
- [82] MINKOV, E., WANG, R. C., AND COHEN, W. W. Extracting personal names from email: applying named entity recognition to informal text. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing HLT/EMNLP, Vancouver, Canada* (2005), Association for Computational Linguistics, pp. 443–450.
- [83] MOONEY, R. J., AND BUNESCU, R. C. Subsequence kernels for relation extraction. In *Advances in Neural Information Processing Systems NIPS, Vancouver, Canada* (2005), pp. 171–178.
- [84] MORO, A., RAGANATO, A., AND NAVIGLI, R. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics* 2 (2014), 231–244.
- [85] NADEAU, D., AND SEKINE, S. A survey of named entity recognition and classification. *Linguisticae Investigationes* 30, 1 (2007), 3–26.
- [86] NADEAU, D., TURNEY, P., AND MATWIN, S. Unsupervised named-entity recognition: Generating gazetteers and resolving ambiguity. 266–277.
- [87] NAVIGLI, R., JURGENS, D., AND VANNELLA, D. Semeval-2013 task 12: Multilingual word sense disambiguation. In *Proceedings of the 7th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2013, Atlanta, USA* (2013), pp. 222–231.
- [88] NAVIGLI, R., LITKOWSKI, K. C., AND HARGRAVES, O. Semeval-2007 task 07: Coarse-grained english all-words task. In *Proceedings of the 4th International Workshop on Semantic Evaluations SemEval@ACL, Prague, Czech Republic* (2007), Association for Computational Linguistics, pp. 30–35.

- [89] NAVIGLI, R., AND PONZETTO, S. P. Babelnet: Building a very large multilingual semantic network. In *Proceedings of the 48th annual meeting of the Association for Computational Linguistics, Sweden* (2010), Association for Computational Linguistics, pp. 216–225.
- [90] NOTHMAN, J., RINGLAND, N., RADFORD, W., MURPHY, T., AND CURRAN, J. R. Learning multilingual named entity recognition from wikipedia. *Artificial Intelligence* 194 (2013), 151–175.
- [91] PAK, A., AND PAROUBEK, P. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the International Conference on Language Resources and Evaluation, (LREC), Malta* (2010), pp. 1320–1326.
- [92] PAN, S. J., AND YANG, Q. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering* (2010), 1345–1359.
- [93] PEARL, J. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, 1988.
- [94] PHELAN, O., MCCARTHY, K., AND SMYTH, B. Using twitter to recommend real-time topical news. In *Proceedings of the 2009 (ACM) Conference on Recommender Systems, RecSys, New York, USA* (2009), pp. 385–388.
- [95] PRADHAN, S. S., LOPER, E., DLIGACH, D., AND PALMER, M. Semeval-2007 task 17: English lexical sample, srl and all words. In *Proceedings of the 4th International Workshop on Semantic Evaluations SemEval@ACL, Prague, Czech Republic* (2007), Association for Computational Linguistics, pp. 87–92.
- [96] RAMAGE, D., HALL, D., NALLAPATI, R., AND MANNING, C. D. Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, (EMNLP), Singapore* (2009), Association for Computational Linguistics, pp. 248–256.

- [97] RAMSHAW, L. A., AND MARCUS, M. P. Text chunking using transformation-based learning. *CoRR cmp-lg/9505040* (1995).
- [98] RAO, D., MCNAMEE, P., AND DREDZE, M. Entity linking: Finding extracted entities in a knowledge base. In *Multi-source, multilingual information extraction and summarization*. Springer, 2013, pp. 93–115.
- [99] RATINOV, L., ROTH, D., DOWNEY, D., AND ANDERSON, M. Local and global algorithms for disambiguation to wikipedia. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1, Portland, USA* (2011), pp. 1375–1384.
- [100] RAU, L. F. Extracting company names from text. In *Proceedings of the Seventh IEEE Conference on Artificial Intelligence Applications* (1991), vol. 1, pp. 29–32.
- [101] RILOFF, E., JONES, R., ET AL. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence and Eleventh Conference on Innovative Applications of Artificial Intelligence, Florida, USA*. (1999), pp. 474–479.
- [102] RITTER, A., CLARK, S., MAUSAM, AND ETZIONI, O. Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, (EMNLP), Edinburgh, UK* (2011), pp. 1524–1534.
- [103] RITTER, A., ETZIONI, O., CLARK, S., ET AL. Open domain event extraction from twitter. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge Discovery and Data Mining, Beijing, China* (2012), pp. 1104–1112.
- [104] RIZZO, G., CANO, A. E., PEREIRA, B., AND VARGA, A. Making sense of microposts (# microposts2015) named entity recognition & linking challenge. pp. 44–53.

- [105] RIZZO, G., AND TRONCY, R. Nerd: evaluating named entity recognition tools in the web of data.
- [106] RIZZO, G., AND TRONCY, R. Nerd: a framework for unifying named entity recognition and disambiguation extraction tools. In *13th Conference of the European Chapter of the Association for Computational Linguistics (EACL), France* (2012), pp. 73–76.
- [107] RIZZO, G., VAN ERP, M., PLU, J., AND TRONCY, R. Making sense of microposts (#microposts2016) named entity recognition and linking (NEEL) challenge. In *Proceedings of the 6th Workshop on 'Making Sense of Microposts' co-located with the 25th International World Wide Web Conference (WWW), Montréal, Canada.* (2016), pp. 50–59.
- [108] RIZZO, G., VAN ERP, M., AND TRONCY, R. Inductive entity typing alignment. In *Proceedings of the Second International Workshop on Linked Data for Information Extraction (LD4IE) co-located with the 13th International Semantic Web Conference (ISWC), Riva del Garda, Italy* (2014), pp. 55–66.
- [109] ROSENBLATT, F. Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Tech. rep., Berlin, Heidelberg, 1961.
- [110] SAKAKI, T., OKAZAKI, M., AND MATSUO, Y. Earthquake shakes twitter users: real-time event detection by social sensors. In *Proceedings of the 19th International Conference on World Wide Web, (WWW) 2010, Raleigh, North Carolina, USA* (2010), pp. 851–860.
- [111] SETTLES, B. Biomedical named entity recognition using conditional random fields and rich feature sets. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications, Geneva, Switzerland* (2004), Association for Computational Linguistics, pp. 104–107.
- [112] SEYMORE, K., MCCALLUM, A., AND ROSENFELD, R. Learning hidden markov model structure for information extraction. In *AAAI-99 Workshop on Machine Learning for Information Extraction* (1999), pp. 37–42.

- [113] SHEN, W., WANG, J., LUO, P., AND WANG, M. Linking named entities in tweets with knowledge base via user interest modeling. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, USA* (2013), pp. 68–76.
- [114] SHI, F., LI, J., TANG, J., XIE, G., AND LI, H. Actively learning ontology matching via user interaction. In *8th International Semantic Web Conference, (ISWC), Chantilly, USA* (2009), Springer, pp. 585–600.
- [115] SHINYAMA, Y., AND SEKINE, S. Named entity discovery using comparable news articles. In *Proceedings of the 20th International Conference on Computational Linguistics COLING, Geneva, Switzerland* (2004), Association for Computational Linguistics, p. 848.
- [116] SNYDER, B., AND PALMER, M. The english all-words task. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text, Barcelona, Spain* (2004), pp. 41–43.
- [117] STOILOS, G., STAMOU, G., AND KOLLIAS, S. A string metric for ontology alignment. In *Proceedings of the 4th International Semantic Web Conference, (ISWC), Galway, Ireland* (2005), Springer, pp. 624–637.
- [118] SUN, Y., MA, L., AND WANG, S. A comparative evaluation of string similarity metrics for ontology alignment. *JOURNAL OF INFORMATION & COMPUTATIONAL SCIENCE* 12, 3 (2015), 957–964.
- [119] SUTTON, C., AND MCCALLUM, A. *An introduction to conditional random fields for relational learning*, vol. 2. 2006.
- [120] TAKEUCHI, K., AND COLLIER, N. Bio-medical entity extraction using support vector machines. *Artificial Intelligence in Medicine* 33, 2 (2005), 125–137.

- [121] THALHAMMER, A., AND RETTINGER, A. Browsing dbpedia entities with summaries. In *European Semantic Web Conference (ESWC) Satellite Events* (2014), Springer, pp. 511–515.
- [122] THOMPSON, C. A., CALIFF, M. E., AND MOONEY, R. J. Active learning for natural language parsing and information extraction. In *Proceedings of the Sixteenth International Conference on Machine Learning (ICML), Bled, Slovenia* (1999), Citeseer, pp. 406–414.
- [123] TJONG KIM SANG, E. F., AND DE MEULDER, F. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh Conference on Natural Language Learning (CoNLL) , Held in cooperation with HLT-NAACL, Edmonton, Canada* (2003), Association for Computational Linguistics, pp. 142–147.
- [124] TUMASJAN, A., SPRENGER, T. O., SANDNER, P. G., AND WELPE, I. M. Predicting elections with twitter: What 140 characters reveal about political sentiment. 178–185.
- [125] TURIAN, J., RATINOV, L., AND BENGIO, Y. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the 48th annual meeting of the Association for Computational Linguistics ACL, Uppsala, Sweden* (2010), pp. 384–394.
- [126] WANG, H., CAN, D., KAZEMZADEH, A., BAR, F., AND NARAYANAN, S. A system for real-time twitter sentiment analysis of 2012 us presidential election cycle. In *The 50th Annual Meeting of the Association for Computational Linguistics (ACL), Proceedings of the System Demonstrations, Jeju Island, Korea* (2012), Association for Computational Linguistics, pp. 115–120.
- [127] WANG, Y., LIU, W., AND BELL, D. A concept hierarchy based ontology mapping approach. In *Proceedings of the 4th International Conference on Knowledge Science, Engineering and Management (KSEM), Belfast, UK* (2010), Springer, pp. 101–113.

- [128] WENG, J., AND LEE, B.-S. Event detection in twitter.
- [129] YAKUSHIJI, A., TATEISI, Y., MIYAO, Y., AND TSUJII, J. Event extraction from biomedical papers using a full parser. In *Proceedings of the 6th Pacific Symposium on Biocomputing PSB, Hawaii, USA* (2001), vol. 6, pp. 408–419.
- [130] YAMADA, I., TAKEDA, H., AND TAKEFUJI, Y. An end-to-end entity linking approach for tweets. In *Proceedings of the the 5th Workshop on Making Sense of Microposts co-located with the 24th International World Wide Web Conference (WWW), Florence, Italy* (2015), pp. 55–56.
- [131] YAMADA, I., TAKEDA, H., AND TAKEFUJI, Y. Enhancing named entity recognition in twitter messages using entity linking. *ACL-IJCNLP* (2015), 136.
- [132] YANG, Y., AND CHANG, M.-W. S-mart: Novel tree-based structured learning algorithms applied to tweet entity linking. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL, Beijing, China* (2015), pp. 504–513.
- [133] ZELENKO, D., AONE, C., AND RICHARDELLA, A. Kernel methods for relation extraction. *Journal of Machine Learning Research* 3, Feb (2003), 1083–1106.
- [134] ZHOU, G., AND SU, J. Named entity recognition using an hmm-based chunk tagger. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, (ACL), Philadelphia, USA* (2002), pp. 473–480.
- [135] ZHU, X., AND GOLDBERG, A. B. Introduction to semi-supervised learning. *Synthesis lectures on Artificial Intelligence and Machine Learning* 3, 1 (2009), 1–130.