ONTOLOGY EXTRACTION FROM CROSS-GENRE UNSTRUCTURED TEXT

A thesis

Submitted in Partial Fulfillment of the Requirement for the Degree of

Master of Computer Science and Engineering

Of

Jadavpur University

By

Promita Maitra
Registration No.: 129002 of 2014-15
Examination Roll No.: M4CSE1616

Under the Guidance of

Dr. Dipankar Das

Department of Computer Science and Engineering
Jadavpur University, Kolkata-700032
India
May, 2016

ONTOLOGY EXTRACTION FROM CROSS-GENRE UNSTRUCTURED TEXT

A thesis

Submitted in Partial Fulfillment of the Requirement for the Degree of

Master of Computer Science and Engineering

Of

Jadavpur University

By

Promita Maitra
Registration No.: 129002 of 2014-15
Examination Roll No.: M4CSE1616

Under the Guidance of

Dr. Dipankar Das

Department of Computer Science and Engineering
Jadavpur University, Kolkata-700032
India
May, 2016

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

Certificate of Recommendation

This is to certify that the dissertation entitled **"Ontology Extraction from Cross-Genre Unstructured Text"** has been carried out by **Promita Maitra** *(University Registration No.: 129002 of 2014-15, Examination Roll No.: M4CSE1616)* under my guidance and supervision and can be accepted in partial fulfillment of the requirement for the Degree of Master of Computer Science and Engineering. The research results presented in the thesis have not been included in any other paper submitted for the award of any degree in any other University or Institute.

……

Dr. Dipankar Das
(Thesis Supervisor)

Department of Computer Science and Engineering

Jadavpur University, Kolkata-32

Countersigned

……
Prof. Debesh Kumar Das

Head, Department of Computer Science and Engineering,

Jadavpur University, Kolkata-32.

……

Prof. Sivaji Bandyopadhyay

Dean, Faculty of Engineering and Technology,

Jadavpur University, Kolkata-32.

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

Certificate of Approval*

This is to certify that the thesis entitled "**Ontology Extraction from Cross-Genre Unstructured Text**" is a bona-fide record of work carried out by **Promita Maitra** in partial fulfillment of the requirements for the award of the degree of Master of Computer Science and Engineering in the Department of Computer Science and Engineering, Jadavpur University during the period of June 2015 to May 2016. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the thesis only for the purpose for which it has been submitted.

……..

Signature of Examiner 1

Date:

……..

Signature of Examiner 2

Date:

*Only in case the thesis is approved

FACULTY OF ENGINEERING AND TECHNOLOGY

JADAVPUR UNIVERSITY

Declaration of Originality and Compliance of Academic Ethics

I hereby declare that this thesis entitled "**Ontology Extraction from Cross-Genre Unstructured Text**" contains literature survey and original research work by the undersigned candidate, as part of her Degree of Master of Computer Science & Engineering.

All information have been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, I have fully cited and referenced all materials and results that are not original to this work.

Name: **Promita Maitra**

Registration No: 129002 of 2014-15

Exam Roll No.: M4CSE1616

Thesis Title: Ontology Extraction from Unstructured Text

…..……..

Signature with Date

Acknowledgement

I would like to express my heartfelt gratitude to my advisor, **Dr. Dipankar Das**, Assistant Professor, Department of Computer Science and Engineering, Jadavpur University for his guidance from the inception of this project and for his patience and constant encouragement throughout the research period.

I would also like to thank **Prof. Debesh Kumar Das**, Head, Department of Computer Science and Engineering, Jadavpur University and **Prof. Sivaji Bandyopadhyay**, Dean, Faculty of Engineering and Technology, Jadavpur University for providing me with their support and required resources to carry out this work.

All the members of NLP lab, specially my seniors *Braja Gopal Patra* and *Tapabrata Mondal*, have been very supportive at times of need. Brilliant suggestions from many of my friends have helped me to solve various issues I have faced during this journey. I would specially like to mention about *Srijita Basu* and *Souvick Ghosh* for their help and for being a source of inspiration.

This thesis would not have been completed without the inspiration and support of a number of wonderful individuals — my thanks and appreciation to all of them for being part of this journey and making this thesis possible.

……..

Promita Maitra

Registration No: 129002 of 2014-15

Exam Roll No.: M4CSE1616

Department of Computer Science & Engineering, Jadavpur University

Table of Contents

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

The rapid growth in the amount of digitalized texts in recent years (specially, in the fields including scientific, clinical, enterprise, legal, and personal information management) has made the management of textual information increasingly important. Moreover, increasing access to internet gives rise to a continuous flow of user generated contents in a large scale for web platforms like Twitter or Blog. However, information is only valuable to the extent that it is accessible, easily retrieved and concerns the personal interests of the user. The growing volume of data, the lack of structured information, and the information diversity have made information and knowledge management a real challenge. Though the way we can guarantee maximum accuracy is to appoint human experts for the task, it is next to impossible to do this manually as it requires magnanimous time and effort. Hence we need automatic systems that are able to perform unstructured text analysis and useful information extraction. Natural Language Processing (NLP) provides the foundation of several technologies related to the management of text information. Ontology engineering is one such study where we try to build a system using text analysis and machine learning tools which can automatically infer knowledge from unstructured text.

1.1 What is Ontology?

The concept of ontology, in the area of artificial intelligence, is defined as the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary (Neches et al., 1991). And the most widely accepted definition of ontology is a "formal, explicit specification of a shared conceptualization" (Gruber, 1993). In simple terms, ontology is a standard structured representation of knowledge acquired from a specific domain. The main components of ontology are domain-terms or concepts and relations among them. Relations can be of two types: taxonomic and non-taxonomic. The relations among entities which hold the entities in hierarchical manner, such as is_a or type_of etc. are called taxonomic relation. On the other hand, the relations which cannot be represented in a tree structure and are often domain-specific, are called non-taxonomic

relations, such as verb based relation or entity-dependent relations like *organization-<in>-location* etc.

Ontology defines a common vocabulary for researchers who need to share information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them. Why would someone want to develop ontology? Some of the reasons are:

- *To share common understanding of the structure of information among people or software agents*
- *To enable reuse of domain knowledge*
- *To make domain assumptions explicit*
- *To separate domain knowledge from the operational knowledge*
- *To analyze domain knowledge*

Technically, researchers use ontologies to describe the semantics of websites. The W3C defines the Semantic Web as "the abstract representation of data on the World Wide Web, based on the RDF standards and other standards to be defined" and has been developing it in collaboration with many researchers and organizations. A new generation of Semantic Web will not only be for human but also for the computer (information agents) to bring semantic content, so that the computer (or the information agent) can "understand" web content, so as to realize the automation of information processing.

Ontologies can broadly be categorized into two types: Domain Ontology and Upper Ontology.

- ***Domain Ontology:***

A domain ontology (or domain-specific ontology) represents concepts which belong to part of the world. Particular meanings of terms applied to that domain are provided by domain ontology. For example, the word card has many different meanings. An ontology about the domain of poker would model the "playing card" meaning of the word, while an ontology about the domain of computer hardware would model the "punched card" and "video card" meanings. Since domain ontologies represent concepts in very specific and often eclectic

ways, they are often incompatible. As systems that rely on domain ontologies expand, they often need to merge domain ontologies into a more general representation. This presents a challenge to the ontology designer. Different ontologies in the same domain arise due to different languages, different intended usage of the ontologies, and different perceptions of the domain by developer or user.

- ***Upper Ontology*:**

An upper ontology (or foundation ontology) is a model of the common objects that are generally applicable across a wide range of domain ontologies. It usually employs a core glossary that contains the terms and associated object descriptions as they are used in various relevant domain sets.

1.2 Properties and Applications of Ontology

Often building an ontology of the domain is not a goal in itself. Developing an ontology is analogous to defining a set of data and their structure for other programs or applications to use. Problem-solving methods, domain-independent applications, and software agents use ontologies and knowledge bases built from ontologies as data. In order to appear as efficient knowledge source, ontologies should exhibit certain properties. They are:

Coverage: All possible domain concepts must be there; the knowledge base must be sufficiently populated depending on the application requirement. Tools are needed to extensively support the task of identifying the relevant concepts and the relations among them.

Consensus: Decision making is a difficult activity for one person, and it gets even harder when a group of people must reach consensus on a given issue and, in addition, the group might need to work from different locations. When a group of enterprises decide to cooperate in a given domain, they first need to agree on many basic issues; that is, they must reach a consensus of the business domain. Such a common view must be reflected by the domain ontology.

Accessibility: The ontology that has been built must be easily accessible: tools are needed to easily integrate the ontology within an application that may clearly show.

Wide use of ontologies these days includes, but not restricted to these applications:

- *Knowledge representation and knowledge management systems*

Knowledge management (KM) is the process of capturing, developing, sharing, and effectively using organizational knowledge. It refers to a multi-disciplinary approach to achieving organizational objectives by making the best use of knowledge. Knowledge representation and reasoning (KR) is the field of artificial intelligence (AI) dedicated to representing information about the world in a form that a computer system can utilize to solve complex tasks.

- *Intelligent query-answering systems*

The process of intelligent query answering consists of analyzing the intent of query, rewriting the query based on the intention and other kinds of knowledge, and providing answers in an intelligent way [Lin et al, 2004]. Intelligent answers could be generalized, neighborhood or associated information relevant to the query. Knowledge, either intentional or extensional, is the key ingredient of intelligence. Many researchers propose to integrate data mining techniques as a knowledge discovery engine to serve an intelligent query answering purpose.

- *Information retrieval and extraction*

Information retrieval is the activity of obtaining information resources relevant to an information need from a collection of information resources. Information Extraction (IE) is the task of automatically extracting structured information from unstructured and/or semi-structured machine-readable documents. Having appropriate domain ontology will speed up these tasks manifold.

- *Semantic Web*

Semantic web is an effort to enhance current web so that computers can process the information presented on WWW, interpret and connect it, to help humans to find required knowledge. Semantic web is intended to form a huge distributed knowledge based system. The focus of semantic web is to share data instead of documents. In other words, it is a project that should provide a common framework that allows data to be shared and reused across application, enterprise, and community boundaries. To facilitate this process, different formats have been

developed as standards for the sharing and integration of data and knowledge—the latter in the form of rich conceptual schemas called ontologies.

1.3 Subtasks in Ontology Extraction

Ontology engineering is basically a combination of several subtasks which themselves are important research challenges in the field of Natural Language Processing. What we need to keep in mind before discussing those steps in ontology creation is that:

- There is not a correct way to model a domain: one person's idea of the domain will for sure differ from another. Developers need to discuss it and reach an agreement, even when none of the individual concepts is essentially wrong.

- The ontology development is an iterative process. It needs constant monitoring and updating with new information that arises or based on new application to be built.

1.3.1 Entity Extraction

Entity extraction is a widely acknowledged task in the field of Text Analysis involving automatic discovery of domain specific terms or phrases and categorizing them into some predefined classes or entity types. In case of ontology engineering, researchers have focused on extracting domain-specific entities. Domain specific entities can include any noun phrase, and thereafter some instances of the entity types are collected for ontology population. For example, some of the entities of an academic institution can be Staffs, Buildings, Departments etc. Within a single entity type, there can also be another or more layers of entity types. In the previous example, 'Staffs' is an umbrella term to act as an upper layer for nested layers like 'Professor', 'Librarian', 'Principal' etc. Also, there can be a few broadly generalized classes called as 'Top Level Classes' which in turn, contain these 'Domain Specific Classes'. Examples of these top level classes can be 'Person', 'Organization', 'Location', 'Time' etc. An efficient ontology engineering system will try to identity as many entities as possible, and classify them correctly according to these pre-decided entity-types. Generally, to come up with specific domain related entities, suggestions from one domain expert can prove to be crucial and helpful. Moreover, during the ontology population stage as well, constant supervision of a domain expert is advisable so as to determine the quality and accuracy of the entities extracted.

1.3.2 Relation Extraction

Ontologies are carefully designed to cultivate the domain at hand and along with the entities detected, we also need to have an understanding of how they are related. Hence, this field of relation extraction came into picture with gigantic research scope. There can be many factors associated with understanding the meaning of an unstructured text. The difficulty lies both in identifying those factors and then coming up with algorithms and implementing methods to handle those factors effectively in order to achieve a consensus upon retrieved relations among entities.

- ***Taxonomic Relation Extraction:***

  In this step, the ontology system tries to arrange the extracted concepts in a taxonomic structure. This can be achieved by unsupervised hierarchical clustering methods. Because the result of such methods is often noisy, a constant supervision, e. g. by evaluation by the domain expert, is integrated. A further method for the derivation of a concept hierarchy exists in the usage of several patterns, which should indicate a sub- or supersumption relationship. Patterns like "X, that is a Y" or "X is a Y" indicate that X is a subclass of Y. Such pattern can be analyzed efficiently, but the recall suffers as frequency of appearance of these patterns is pretty low. Instead of manually extracting these patterns from domains, bootstrapping methods are developed, which learn these patterns automatically and therefore ensure a higher coverage.

- ***Non-taxonomic Relation Extraction:***

  Non-taxonomic relations are those for which entities cannot be represented in a hierarchical fashion based on the relations among them. It is argued to be as one of the most difficult task and often neglected problem in ontology learning mechanism. It is this kind of relations which reveal more about a particular domain as taxonomic relations are restricted to some specific relations only, causing a hindrance to explore the domain in an exhaustive manner. As these relations vary immensely owing to the diverse nature of domains to extract an ontology from, it is very difficult to figure out how many type of relations are there to be extracted. The problem of non-taxonomic relation extraction can be categorized into two sub-problems:

(a) Non-taxonomic Relation Discovery: Identification of the domain concept pairs (C1, C2) such that some non-taxonomic relations hold from C1 → C2 or/and C2 → C1.

(b) Non-taxonomic Relation Labeling: Identification of labels for the non-taxonomic relations from C1 → C2 or/and C2 → C1.

1.4 Thesis Challenge

Ontologies are the structured representation of information from a specific domain. The main challenge of our present task is twofold.

First, we are dealing with texts of different genre; which essentially means that the datasets we have considered differs significantly in terms of text structure, vocabulary used, influence of external agents like region, time, culture, platform etc. Text snippets from web-based platforms like Twitter, Blog or Review data are similar in nature, albeit not exactly same. Similarly, the characteristics of texts from Wikipedia articles are close in nature with those of the NEWS article. However, mythological corpus and contemporary literature corpus exhibits completely different literary style.

Contrary to popular belief, the ontology we will be constructing will not be on a particular theme as the texts of the datasets we are considering can be on any random topic or theme. For example, the tweets from the domain twitter can be on different topics and can contain various types of concepts depending upon the topic it is discussing, which makes it practically impossible to have domain specific concepts or entity types. Hence, we need to consider only the top-level entity types like PERSON, ORGANIZATION or LOCATION where a PERSON entity can contain instances from any domain specific person roles like Student, Officer, Doctor, President etc. We face a similar problem in case of non-taxonomic relation extraction and classification as well. Instead of having domain specific relations like <Doctor>-treats-<Patient> or <Employee>-works in-<Organization>, we need to rely on more abstract relations.

1.5 Thesis Contribution

The main aim of present work is to build a system that can extract ontological information from cross-genre unstructured text, which can practically hold information about any random domain. As the datasets we have considered exhibit prominent variance in terms of lexical and syntactic structure, the main challenge is to build a system that will be abstract enough to perform on all types of data, yet efficient enough to be able to extract meaningful information as well. An entity identification and classification system is built that recognizes and classifies top-level named entities from texts. Next, we try to find out the possible relations that can exist among these entities. As we have only considered named entities, it practically doesn't make sense to try to extract direct taxonomic relations among them. Hence, we built a module that assigns scores to each pair of entities for six different taxonomic relations (Synonym, Antonym, Hypernym, Hyponym, Holonym, Meronym) based on context word analysis. We have proposed two different non-taxonomic relation extraction schemes: one is verb based, and another is entity-pair sentiment based. In the verb based relation extraction module, we extract relations based on entity co-occurrence and the verb present, further filtering and clustering the extractions based on thematic role and selectional restrictions from VerbNet. In sentiment based relation module, we try to predict a polarity label for each pair on entity that co-occurs, based on the context words of co-occurrence.

1.6 Introduction to Later Chapters

Chapter 2 describes the source, nature and appropriate statistics of the datasets we have considered for our work with samples from each domain. Chapter 3 contains in detail our approach to the entity extraction and classification task with evaluation results and observations. In Chapter 4, we explain our taxonomic relation scoring system with sample results and observations. Chapter 5 is about the non-taxonomic relations that we propose to extract from these cross-genre and/or multi-domain texts. Detailed description of our modules, along with sample relations extracted and result statistics is presented. In Chapter 6, we conclude the work done and list down the future directions of present challenge.

CHAPTER 2

DATASET PREPARATION

For any kind of experiment to yield effective results, we need to collect or build a dataset free from any type of bias or irregularity. Ontologies are generally built for specific domains or topics like Marine Biology, Chinese Food, and Academic Institution etc. But in our approach, we shifted our focus from topic to genre while collecting the datasets. To carry on the task at hand successfully, we needed to have unstructured textual data from different genre of writing such as social media texts, mythological texts etc., which in turn, can talk about multiple topics in single dataset The complexity of our domain adaptation problem is hence double fold. First, it does not revolve around a particular theme or topic; and secondly, it does not deal with a regular structure of data. Initially, we planned to apply our system in two different domains with major lexical difference such as mythological text and contemporary literature. However, we noticed that the social media texts of recent times can contribute some important insights in this regard as they are vastly irregular in structure, follows no standard grammatical rules, incorporates various foreign words and symbols and emoticons. These unusual characters make it quite difficult to extract relevant information from these kinds of noisy texts. One can even notice more than one genre among these texts as well, such as Social Networking (Twitter/Facebook) data, Blog data, Review data, SMS data etc. As each of them serves different purposes, the way they are written also changes significantly. For our work, we have mainly considered data from three domains: Twitter, Blog and Review. On the other hand, the mythological texts that we are considering diverges manifold from contemporary style of writing that we are acquainted to. Hence, we have considered both of them to observe the performance of our system in these two vastly different genres of texts. Lastly, a wiki article corpus and one news corpus have also been included in our dataset as news and Wikipedia articles ideally follow a neutral news reporting fashion of writing without many regional or cultural influences.

Second thing that we needed to keep in mind is that we needed data from which we can extract similar entities and entity-pair relations. To maintain this uniformity, we need to choose utterly generalized types of entities that can be found in all these domains of texts. As none of them deals about a specific prefixed topic, there is no way we can come to a consensus about the

domain-specific entity types. So while collecting and assembling the data, we needed to keep these two points in mind.

2.1 Dataset Sources & Description

Following is a brief description of the sources and formats of our entire dataset:

2.1.1 Twitter Data

The organizers of Microposts-2016 (Rizzo et al., 2016) workshop conducted one shared task titled Named Entity Extraction and Linking (NEEL), which was a part of World Wide Web (WWW)-2016 conference. We participated in the said shared task and received the Named Entity annotated tweet dataset for training our model. The dataset consists of tweets extracted from a collection of over 18 million tweets. The dataset includes event-annotated tweets provided by the Redites project (http://demeter.inf.ed.ac.uk/redites/) covering multiple noteworthy events from 2011, 2013 (including the death of Amy Winehouse, the London Riots, the Oslo bombing and the Westgate Shopping Mall shootout), tweets extracted from the Twitter firehose from 2014 and 2015 via a selection of hash-tags. Since the task of this challenge was to automatically recognize and link entities, they have built the dataset considering both event and non-event tweets. While event tweets are likely to contain entities, non-event tweets were also kept to evaluate the performance of the system in avoiding false positives in the entity extraction phase. The training set is built on top of the entire corpus of the previous years' NEEL Challenges.

The Gold Standard was generated with the help of 3 annotators. The annotation process followed three phases. In the first one, an unsupervised annotation of the Gold Standard has been performed, with the intent to extract candidate links which were meant as inputs of the second stage.

In the second stage annotations were performed by two annotators using GATE. The annotators were asked to analyze the entity mentions, categories and links provided in the first stage and to add, remove any others. The annotators were also asked to mark any problematic case if encountered.

In the third phase, a third annotator went through the problematic cases and, involving the two initial annotators, refined the annotation procedures. An iterative process has then taken place looping on stage 2 and 3, till mostly all problematic cases were resolved.
They came up with 7 different entity types in total: *PERSON*, *LOCATION*, *ORGANIZATION*, *THING*, *EVENT*, *CHARACTER* and *PRODUCT*. The taxonomy of possible entity types that these labels should cover is as follows:

- *THING-*
  - Languages, ethnic groups, nationalities, religions, diseases, sports, astronomical objects
- *EVENT-*
  - Holidays, sport events, political events, social events
- *CHARACTER-*
  - fictional character, comics character, title character
- *LOCATION*
  - public places (squares, opera houses, museums, schools, markets, airports, stations, swimming pools, hospitals, sports facilities, youth centers, parks, town halls, theatres, cinemas, galleries, universities, churches, medical centers, parking lots, cemeteries)
  - regions (villages, towns, cities, provinces, countries, continents, dioceses, parishes)
  - commercial places (pubs, restaurants, depots, hostels, hotels, industrial parks, nightclubs, music venues, bike shops)
  - buildings (houses, monasteries, creches, mills, army barracks, castles, retirement homes, towers, halls, rooms, vicarages, courtyards)
- *ORGANIZATION*
  - companies (press agencies, studios, banks, stock markets, manufacturers, cooperatives)
  - subdivisions of companies
  - brands
  - political parties
  - government bodies (ministries, councils, courts, political unions)

- press names (magazines, wiki articlepapers, journals)
- public organizations (schools, universities, charities)
- collection of people (sport teams, associations, theater companies, religious order, youth organizations, musical band)

- *PERSON*
  - People's names (titles and roles are not included, such as Dr. or President)
- *PRODUCT*
  - Movies, tv series, music albums, press products (journals, wiki article papers, magazines, books, blogs), devices (cars, vehicles, electronic devices), operating systems, programming languages

While working on the task, we observed that the entity types Person-Character and Thing-Product are quite ambiguous. Multiple overlapping of these two entity types were detected in the annotated corpus which led us to simplify the entity tagset for our task by keeping a single tag *PERSON* for both *PERSON* and *CHARACTER* names, while *THING* represents all the entities that come under *THING* and *PRODUCT*. So finally for our work, we are left with five named entity types for twitter dataset: *PERSON, LOCATION, ORGANIZATION, THING, and EVENT.*

2.1.2 Blog Dataset

The next type of social media data that we had in mind is texts crawled from blogs. Texts found in blogs do not conform to the style of any particular genre per se varying from person to person, and thus offers a variety in writing styles, choice and combination of words, as well as topics. We used the dataset built by Aman & Szpakowicz for their emotion analysis task on blog posts (Aman and Szpakowicz, 2007) (Aman and Szpakowicz, 2008) excluding the emotions labels associated with that dataset, we have collected the blog posts directly from the web. First, a list of seed words was prepared for six basic emotion classes. Next, using the seed words for each category, they retrieved blog posts containing one or more of those words. As these sentences were not entity annotated, we use the Stanford CoreNLP tagger to annotate the data. But it doesn't provide *THING* or *EVENT* tag. Hence we use Stanford's 3-class model tagging entities

belonging to the classes *PERSON, LOCATION* and *ORGANIZATION*. We use this Stanford NER tagged data as the gold standard for training and evaluation purposes.

2.1.3 Review Dataset

Another kind of user generated texts that are available in web is product or service review. As any NER tagged review corpus was not available for free, we again had no choice but to use Stanford NE tagger on some crawled review data as our gold standard for this domain. So the entity classes we will be having are: *PERSON, LOCATION and ORGANIZATION*. So keeping this in mind, we needed to choose such a dataset which will have enough instances of these types of entities. Movie review corpus seemed to be a good fit to this criterion as they have Actors' or Directors' or Producers' or even Characters' name as PERSON entities, various film-shooting or plot locations for LOCATION-type instances and several ORGANIZATION entities as well. Pang and Lee built a sentiment annotated movie review corpus for their Sentiment based on Subjectivity Analysis task (Pang & Lee, 2004). We used the 2004 release of their dataset named 'polarity dataset v2.0' made public in June, keeping only the unprocessed movie review source files for our work and ignoring the sentiment annotations.

2.1.4 Wikipedia Articles

The types of user generated documents in web we discussed so far follow a particular style of writing which gets affected heavily by various factors such as time, trend, region, culture, language knowledge of user etc. But Wikipedia articles, on the contrary, maintain an impersonal flow of words, with a formal writing approach so that it can be understood by a larger number of people across the globe. Standard and non-complex sentence structure and moderate length of articles make it an unbiased, balanced dataset to work with. We use the NE annotated dataset built and released by Nothman et al. in 2012 for their work of multilingual named entity recognition from Wikipedia. We have only used the English dataset for current work. They developed a hierarchical classification scheme for named entities, extending on the BBN scheme (Brunstein, 2002), and have manually labeled over 4,800 English Wikipedia pages. Their logistic regression classifier for Wikipedia articles uses both textual and document structure features, and achieves a

state-of-the-art accuracy of 95% (coarse-grained) when evaluating on popular articles. Having created their own "Wikipedia gold" corpus (wikigold) by manually annotating 39,000 words of English Wikipedia with coarse-grained NE tags, corroborating the results on newswire, their silver-standard English Wikipedia model outperforms gold-standard models on wikigold by 10% f-score. They labels that they have used for entity annotation are: PERSON, LOCATION, ORGANIZATION and MISCELLANEOUS. However, to maintain the uniformity of our approach, we disregarded the MISCELLANEOUS tag and considered the rest.

2.1.5 NEWS Dataset

Newspapers generally adhere to an expository writing style. In its most ideal form, news writing strives to be intelligible to the majority of readers, engaging, and succinct. Within these limits, news stories also aim to be comprehensive. However, other factors are involved, some stylistic and some derived from the media form. Editorial policies dictate the use of adjectives, euphemisms, and idioms. Newspapers with an international audience, for example, tend to use a more formal style of writing. This writing style encompasses not only vocabulary and sentence structure, but also the way in which stories present the information in terms of relative importance, tone, and intended audience. These characteristics make this a balanced dataset both in terms of entity count and text structure to analyze and extract entity-relationships from. We collected the Press Reportage articles from the Brown Corpus of Standard American English, compiled by W.N. Francis and H. Kucera, Brown University, Providence, RI. The corpus consists of one million words of American English texts printed in 1961. The texts for the corpus were sampled from 15 different text categories to make the corpus a good standard reference. However, these news texts were not named entity tagged and hence we used Stanford Named Entity 3-class tagger to create the gold standard annotated using PERSON, LOCATION and ORGANIZATION tags.

2.1.6 Contemporary Literature Corpus

After the web based and news datasets, we thought of applying our system to a dataset which follows standard writing protocols to evaluate the performance against a different

type of texts. We collected the text corpus from the training set of PAN-2015 Authorship Verification task (Stamatatos et al. 2015). The shared task was focused on the problem of author verification: given a set of documents by the same author and another document of unknown authorship, the task is to determine whether or not the known and unknown documents have the same author. The task was aimed for four languages: English, French, Italian and Dutch; though we have only availed the English dataset for training. It includes written documents from 100 different authors, two documents each. The dataset being cross-genre and cross-topic, contains a large variety of entities for each type; however, they were not annotated and so we again used Stanford CoreNLP tagger to annotate the NEs in the dataset. In addition to these texts, we have collected a few famous short stories considered classics in English literature from web and included them in our dataset as well, after tagging those using Stanford Tagger. So the named entity classes we have for this dataset are: PERSON, ORGANIZATION and LOCATION.

2.1.7 Mythology Corpus

Mythological texts are very different from the kind of writing manner we are familiar with. Long sentences, unusual structure, usage of obsolete words and complex phrases make it increasingly difficult to analyze these texts using the available tools or lexico-syntantic patterns we use for general texts. We found an English version of great Indian epic *Mahabharat* on internet translated from the original Sanskrit text by Kisari Mohan Ganguli and we decided to observe our system's performance on a part of it. We chose a translated text over mythologies or folklores originally written in English because this, written in a different region or time or language, will have a diverse set of entities compared to what we have been encountering till now. We wanted to see if our system will be able to detect entities and relations in these cases. We used 3-class Stanford NE tagger here as well, but it failed to perform agreeably specially for the tags ORGANIZATION and LOCATION. We had to manually filter the correct tags from these two entity sets after the Stanford annotation phase. Finally, we were left with a huge set of PERSON entities, but very few LOCATION and ORGANIZATION entries in comparison.

2.2 Preprocessing

As discussed above, we had named entity annotations of only the Twitter and Wikipedia dataset. In order to evaluate our system's performance we needed a gold standard tagging of entities for all the domains. Stanford CoreNLP, being one of the best Natural Language Processing Tool, came into picture and all other documents Blog, Review, News, Contemporary Literature and Mythology were sent through a NE tagging module, entirely dependent on Stanford tagger. The Stanford Named Entity Recognizer was used to extract the named entities. It is a CRF classifier implementing linear chain Conditional Random Field. We use the 3 class model to extract the named entities belonging to classes LOCATION, PERSON and ORGANIZATION.

Another separate file was created just to note down the words, their original lemmatized form and their corresponding Part of Speech tags from all the datasets. To get this task successfully done, we again took help from the Stanford CoreNLP POS tagger and lemmatizer. Using Stanford CoreNLP lemmatizer, extracted verbs are transformed into the corresponding lemma, so that they can be used to search VerbNet to acquire a greater understanding. Lemmatization usually uses a vocabulary and does morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the lemma. If confronted with the token saw, stemming might return just s, whereas lemmatization would attempt to return either see or saw depending on whether the use of the token was as a verb or a noun. The two may also differ in that stemming most commonly collapses derivationally related words, whereas lemmatization commonly only collapses the different inflectional forms of a lemma.

2.3 Sample Texts & Entity Statistics

The main aim of current work is to develop a system which will perform consistently independent of the genre or topic of text it is processing. In order to achieve that, we collected texts which follow different forms of literary style and deals with a diverse range of topics or themes. As the forms are different, the distribution of entities throughout the text is also irregular. While the Twitter or Wikipedia dataset can be called an entity-rich corpus with more than 10% words as entities, datasets like Blog, Review and Literature have only 2-3% of words

as entities. This makes it difficult to consider one domain as training and a different one for evaluation, as the system will be heavily biased.

Domain	Total Words	#Entities	#Non-entities	%Entities	%Non-entities
BLOG	43660	622	43038	~1.42	~98.58
REVIEW	51175	974	50201	~1.90	~98.10
TWITTER	99629	11010	88259	~11.09	~88.91
WIKIPEDIA	64730	7048	57682	~10.89	~89.11
NEWS	59539	5672	53867	~9.53	~90.47
LITERATURE	64977	1839	63138	~2.83	~97.17
MYTHOLOGY	66387	4849	61538	~7.30	~92.70

Table 2.1 Percentage of Entity/Non-Entity in the Datasets

Next, sample text snippets from all the domains with NE annotations are listed below:

Dataset	Sample Sentences
TWITTER	1. #Christians<***Thing***> are not marginalised in Kwara<***Location***> state<***Location***>, says their Governor Ahmed<***Person***>: Governor Abdulfatah<***Person***> Ahmed<***Person***>, of Kwara<***Location***> State<***Location***> has... http://t.co/U1VtupsdaA 2. @mariastamp1<***Person***> Yeah thanks Maria<***Person***>, we had a lovely time in the university.
BLOG	1. Montmartre<***Person***>, the old bohemian artist district, five minutes from the Sacred Coeur Cathedral <***Organization***>.

	2. A grim war, an era when planned communities built by government scientists promised an idyllic life for Aamericans . and burbank , California<***Location***> , brings to mind the tonight show and the home of nbc <***Organization***>.
NEWS	1. Implementation of Georgia<***Location***>'s automobile title law was also recommended by the outgoing jury of central Court < ***Organization*** >. 2. Miss Katherine Vickery <***Person***>, who attends Sweet Briar College<***Organization***> in Virginia<***Location***>, will rejoin her father , Dr. Eugene Vickery<***Person***> , at the family home in Richmond<***Location***>.
WIKIPEDIA	1. Lincoln <***Person***> was a strong supporter of the American<***Organization***> Whig<***Organization***> version of liberal capitalism. 2. That morning the Princess<***Person***> rose earlier than she had done since she had been carried into Africa<***Location***> by the magician, whose company she was forced to endure once a day.
CONTEMPORARY LITERATURE	1. There's Annette <***Person***>, Olivette <***Person***> and Babette <***Person***>. Three as pretty little French ladies as ever came out of Paris <***Location***>. 2. He told me to tell you he'd be back tomorrow with definite information on IBM<***Organization***> deal.
MYTHOLOGY	1. Vaishampayana<***Person***> said, "O king, the seven Sarasvatis<***Location***> cover this universe! Whithersoever the Sarasvati<***Location***> was summoned by persons of great energy, thither she made her appearance."

Table 2.2 Sample Text Snippets from all Domains

CHAPTER 3

ENTITY EXTRACTION

3.1 Introduction

Entity extraction is a widely acknowledged task in the field of Text Analysis involving automatic discovery of domain specific terms or phrases and categorizing them into some predefined classes or entity types. Entity extraction often serves as a fundamental step for complex Natural Language Processing (NLP) applications such as information retrieval, question answering, machine translation, ontology learning etc. In case of ontology engineering, researchers have focused on extracting domain-specific entities. Domain specific entities can include any noun phrase, and thereafter some instances of the entity types are collected for ontology population. For example, some of the entities of an academic institution can be Staffs, Buildings, Departments etc. Within a single entity type, there can also be another or more layers of entity types. In the previous example, 'Staffs' is an umbrella term to act as an upper layer for nested layers like 'Professor', 'Librarian', 'Principal' etc. Also, there can be a few broadly generalized classes called as 'Top Level Classes' which in turn, contain these 'Domain Specific Classes'. Examples of these top level classes can be 'Person', 'Organization', 'Location', 'Time' etc.

Each of these classes will have ample amount of instances from the unstructured domain text that we are parsing. An efficient ontology engineering system will try to identity as many entities as possible, and classify them correctly according to these pre-decided entity-types. Generally, to come up with specific domain related entities, suggestions from one domain expert can prove to be crucial and helpful. Moreover, during the ontology population stage as well, constant supervision of a domain expert is advisable so as to determine the quality and accuracy of the entities extracted.

Ontology population is generally carried out through some kind of ontology-based information extraction (OBIE). This consists of identifying the key terms in the text (such as named entities and other domain specific technical terms) and then relating them to concepts in the ontology. Typically, the core task of information extraction is carried out by linguistic pre-processing

(tokenization, POS tagging etc.), followed by a named entity recognition component, such as a gazetteer and rule-based grammar or machine learning techniques. Named entity recognition (using such approaches) and automatic domain term recognition are thus generally performed in a mutually exclusive way: i.e. one or other technique is used depending on the ultimate goal. However, it makes sense to use a combination of the two techniques in order to maximize the benefits of both. For example, term extraction techniques are generally built on frequency-based information whereas named entity recognition task usually uses a more linguistic basis. Also to keep in mind that a "term" refers to a specific concept characteristic of a domain, so while a named entity of types such as Person or Location is generic across all domains, a technical term such as "myocardial infarction" is only considered a relevant term when it occurs in a medical domain: if we were interested in building an ontology for sports domain then it would probably not be considered a relevant term, even if it occurred in a sports article. As with named entities, however, terms are generally formed from noun phrases (in some contexts, verbs may also be considered terms, but we shall ignore that in present work).

Term or entity extraction methods can be of two types, either supervised or unsupervised. While the supervised methods make use of an already annotated dataset to train and build an entity recognition system, figuring out the most effective set of features that include word level, orthographic and semantic characteristics of the text fragments. On the other hand, the unsupervised methods mainly depend on statistical and lexical pattern approaches such as co-occurrence, tf-idf measure etc. Recent researches have advanced to address cross-domain, cross-language or even code-mixed entity detection issues.

3.2 Related Work

Named entity recognition (NER) is a technology for recognizing proper nouns (entities) in text and associating them with the appropriate types. Common types in NER systems are location, person name, date, address, etc. Some NER systems are incorporated into Parts-of-Speech (POS) taggers, though there are also many stand-alone applications. Whereas most NER systems are based on analyzing patterns of POS tags, they also often make use of lists of typed entities (like list of possible person names) or regular expressions for particular types (like address patterns). There are three main method of learning NE: Supervised Learning, semi-supervised learning and unsupervised learning. The main shortcoming of supervised learning is the requirement of a large

annotated corpus. The unavailability of such resources and the prohibitive cost of creating them lead to two other alternative learning methods.

Supervised Learning: The idea of supervised learning is to study the features of positive and negative examples of NE over a large collection of annotated documents and design rules that capture instances of a given type. The current dominant technique for addressing the NER problem is supervised learning. SL techniques include Hidden Markov Models (HMM), Decision Trees, Maximum Entropy Models (ME), Support Vector Machines (SVM), and Conditional Random Fields (CRF). These are all variants of the SL approach that typically consist of a system that reads a large annotated corpus, memorizes lists of entities, and creates disambiguation rules based on discriminative features. A baseline SL method, which is often proposed, consists of tagging words of a test corpus, if they are annotated as entities in the training data. The performance of the system depends on the baseline to be transferred to the vocabulary, with the percentage of words that appear without repetition, both in training and test corpus. D. Palmer and Day (1997) calculates the vocabulary transfer to the MUC-6 training data. They report on a transfer of 21%, with as much as 42% of place names not repeated, but only 17% of the organizations and 13% of those names. Vocabulary transfer is a good indicator of the recall (number of people over the total number of units) identifies the baseline system, but is a pessimistic measure, because some bodies are often repeated in the documents. A. Mikheev et al. (1999) is just the recall of the baseline system on the MUC-7 Corpus calculated. They report a recall of 76% for sites, 49% of organizations and 26% for people with precision of 70% to 90%. Whitelaw and Patrick (2003) report consistent results on MUC-7 for the aggregated enamex class. For the three species together, the accuracy of precision 76% and the recall is 48%.

Semi-Supervised Learning: The term "semi-supervision '(or' weak supervision") is still relatively young. The main SSL technology is called "bootstrapping" and includes a small measure of control, like a row of seeds, for the beginning of the learning process. For example, a system aimed at "disease names" could prompt the user to give a small

number of example names. Then the system looks for sentences that contain these names, and tries to identify some clues from the context of five common examples. Then the system tries to other cases of the disease names that appear to be found in similar contexts. The learning curve is then reapplied to the newly found examples, you discover relevant new contexts. By repeating this process, a large number of disease names and a variety of contexts will eventually be obtained. Recent experiments in semi-supervised NER report that rival performances Baseline monitoring approaches.

Unsupervised Learning: The typical approach to unsupervised learning is clustering. For example, one can try to collect names from clustered groups based on the similarity of context. There are other methods also unattended. Basically, the techniques based on lexical resources (eg WordNet), calculated on lexical patterns and statistics on a large unannotated corpus. Here are some examples. E. Alfonseca and Manandhar study (2002), the problem of labeling an input with a corresponding word NE type. NE-types from WordNet (eg taken place> Land, animate "person, animate> Animals, etc.). The approach is to assign a theme to each WordNet synset signature by simply listing words that occur frequently together with him in a large corpus. Then, as a command word will appear in a given document, the word context (words in a fixed-size window around the input word) to the type signature is compared and classified among the similar. Y. Shinyama and Sekine (2004) uses an observation that these bodies often appear simultaneously in several news articles, while not common nouns. You found a strong correlation between a name and unit on time (in time) and simultaneously in multiple news sources. This technique permits the identification of rare proper names in an unsupervised manner and in combination with other useful NER methods.

Domain adaptation has been intensively studied for a variety of sequence labeling tasks in the natural language processing area. Daum´e III & Marcu (2006) proposed to distinguish between general features and domain-specific features by training three separate maximum entropy classifiers. They empirically showed the effectiveness of the proposed method on mention type classification, mention tagging and recapitalization systems. Jiang & Zhai (2007) investigated instance weighting method for semi-supervised domain adaptation by assigning more weights to

labeled source and target data, removing misleading training instances in the source domain, and augmenting target training instances with predicted labels. They empirically evaluated their method for cross domain part-of-speech tagging and named entity recognition to justify its efficacy. Daum´e III (2007) proposed an easy adaptation learning method (EA) by using feature replication, which is later extended into a semi-supervised version (EA++) by incorporating unlabeled data via co-regularization (Daum´e III et al., 2010). These methods demonstrated good empirical performance on a variety of NLP tasks.

Jiang and Zhai (2006) exploit the domain structure contained in the training examples to avoid over-fitting the training domains. Arnold et al. (2008) exploit feature hierarchy for transfer learning in NER. Instance weighting (Jiang and Zhai, 2007) and active learning (Chan and Ng, 2007) are also employed in domain adaptation. Most of these approaches need the labeled target domain samples for the model estimation in the domain transfer. Obviously, they require much effort for labeling the target domain samples. Some approaches exploit the common structure of related problems. Ando et al. (2005) learn predicative structures from multiple tasks and unlabeled data. Blitzer et al. (2006, 2007) employ structural corresponding learning (SCL) to infer a good feature representation from unlabeled source and target data sets in the domain transfer. Guo et al. present LaSA model to overcome the data gap across domains by capturing latent semantic association among words from unlabeled source and target data (Guo et al., 2009). In addition, Miller et al. (2004) and Freitag (2004) employ distributional and hierarchical clustering methods to improve the performance of NER within a single domain. Li and McCallum (2005) present a semi-supervised sequence modeling with syntactic topic models.

3.3 Named Entity rEcognition and Linking Challenge

In present day world, the relevance and importance of various social media platforms are immeasurable. Microposts such as tweets are limited in number of characters. However, the conciseness of the text is barely a pointer to its usefulness. From opinion mining during political campaigns to live feeds during sports events, from product reviews to vacation posts, Twitter is almost ubiquitous. Twitter promotes instant communication. Most celebrities use it to form their own digital presence. It also serves as a common forum where people have the capability to rise from obscurity to prominence through sharing of opinions. If we compare microposts to any standard long document such as blog or news articles, there exist a number of differences. Long

articles are usually well written. They follow a definite structure, include headings and follow the rules of English grammar. Microposts, on the other hand, are short, noisy and hardly show any adherence to formal grammar. Presence of extraneous characters like hashtags, abbreviations and the lack of structure and context makes it difficult to extract relevant information. Due to this complexity, existing named entity recognition systems (NER) do not perform very well with tweet data. In NEEL challenge (Rizzo & Erp, 2016) of #Microposts2016, we were required to automatically identify the named entities and their types from Twitter data and link them to the corresponding URIs of the DBpedia 2015-04 dataset[1]. Identifying the named entities and linking them to an existing knowledge base enriches the text with more contextual and semantic information. The mentions which could not be linked to any existent DBpedia resource page were recognized as NIL mentions. These mentions were clustered to ensure that the same entity, which does not have a corresponding entry in DBpedia, will be referenced with the same NIL identifier. We have developed three systems for the NEEL challenge, the major difference between the systems being the features used for each run. Our system follows a hybrid approach where Stanford Named Entity Recognition system is used to identify the entity mentions. In the next step, we run ARK Twitter Part-of-Speech Tagger to identify the mentions which are missed formerly. We use our own classifier to detect the type of the mentions. The named entity linking to DBpedia resources is done using Babelfy[2] . It must be noted that we followed a feature-based approach for the NEEL challenge. We also combined the existing tools for Named Entity Recognition and Linking. Each of the existing tools, like the Stanford NER, ARK Part-of-Speech Tagger and Babelfy are state-of-the-art. We explored their strengths and weaknesses in our work.

A. System Description

Our system follows four steps in pipeline as shown in Figure 1. Mention detection in two stages, followed by mention type classification, mention linking and NIL clustering.

[1] http://wiki.dbpedia.org/dbpedia-data-set-2015-04

[2] http://babelfy.org/

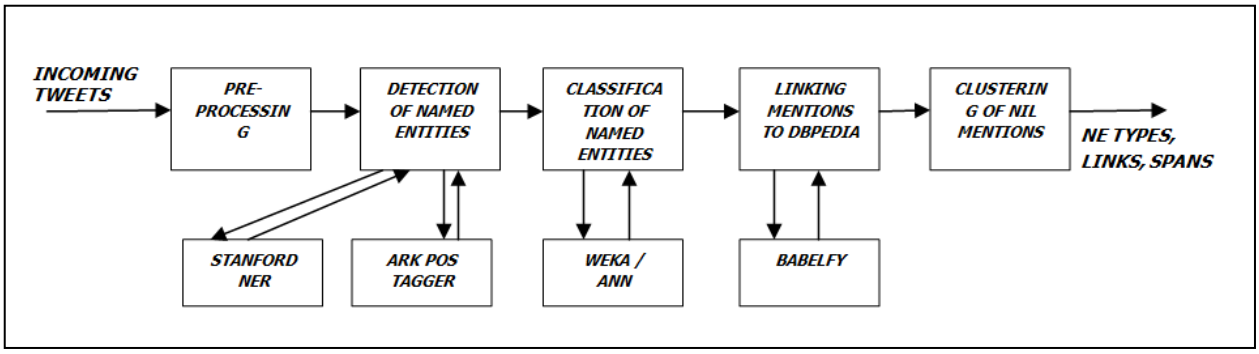

Fig 3.1 System Architecture for NEEL Challenge

Preprocessing

From the training data, the mentions referring to the 7 types of entities were extracted to form 7 bags of words. Using the initial words as seeds, the Wikipedia dumps were crawled to expand the set of words. These lists represent potential candidates for Named Entity mentions.

Detection of Entity Mentions

In this step, the named entity mentions in the given tweets are identified using two different approaches.

i. Using Stanford Named Entity Recognizer

The Stanford Named Entity Recognizer[3] was used to extract the named entities. It is a CRF classifier implementing linear chain Conditional Random Field. We use the 3 class model to extract the named entities belonging to classes Location, Person and Organization. While the recall was very low, the precision of Stanford NER was quite good.

ii. Using ARK Twitter Part-of-Speech Tagger

The tweets were tokenized and assigned Part of Speech tags using the ARK Twitter Part-of-Speech Tagger (Gimpel et al., 2011). We used the Twitter POS model with 25-tag tagset. The proper nouns (NNP and NNS tagged as ^) and possessive proper nouns (tagged Z) along with hashtags (#) and at-mentions (@) were extracted as probable

[3] http://nlp.stanford.edu/software/CRF-NER.shtml

candidates for Named Entity mentions. The mentions which were already identified using Stanford NER are not considered for classification step as they are already classified by the tagger itself. The rest of the mentions are classified using our classifier in the next step.

Classification of Entity Types

In the machine learning software WEKA (Hall et al., 2009), we use the following features to form a feature set and used the Random Forest classifier to generate a pruned C4.5 Decision Tree for 7-way classification of the named entity mentions i.e. Thing, Event, Character, Location, Organization, Person and Product, while providing the identified noun entities from previous steps as input. We checked the accuracy by using various classifiers like Naive Bayes, k-Nearest Neighbour and Support Vector Machine on training data with a 10-fold cross validation. Random Forest gave the best results.

Run 1

The features used for Run 1 were:

Length of the mention string, If the mention is all capitalized, If the mention contains mixed case, If the mention contains digits, If internal period is present in mention string, If present in list of Persons, If present in list of Things, If present in list of Events, If present in list of Characters, If present in list of Locations, If present in list of Organizations, If present in list of Products.

The above-mentioned lists are basically the bag of words produced from the training data in the pre-processing step.

Run 2

We made use of various text based features and bag of words in Run 1. In Run 2, we explored various contextual features in addition to the features of Run 1. So we combined ten new features with the previous twelve features for Run 2. The ten additional features used in Run 2 were as follows:

Context score for Person entity, Context score for Location entity, Context score for Character entity, Context score for Organization entity, Context score for Event entity, Context score for Thing entity, Context score for Product entity, Frequency of Part-of-speech of mention, Frequency of previous Part-of-speech, Frequency of next Part-of-speech.

Context score of a particular mention is calculated for a three word window of the mention. For each class, we have the number of occurrences of each word in a three word window. While calculating the feature value, we assign the sum of the frequency of the words forming that fixed-size window as the mention's context score.

Run 3

We wanted to apply a Feed-Forward neural network (also called the back-propagation networks and multilayer perceptron) to our feature set and see how it performs as these kind of Artificial Neural Networks are useful in constructing a function where the complexity of the feature values makes the decision for building such a function by hand almost impossible. We took the same features of Run 2 and employed a feed-forward neural network based regression model with 5 hidden layers. For the previous two runs, i.e. Run1 and Run2, the tags from Stanford NER were considered as the primary influence over our classifier tags as its accuracy was quite good. For Run 3 however, we omit the Stanford NER influence and let only the neural network model do the tagging to check the efficiency of our classifier.

Linking Mentions to DBpedia

We used the Babelfy java API service (Moro et al., 2014) to address the task of entity linking to DBpedia 2015-04 resources. It is a unified, multilingual, graph-based approach to Entity Linking and Word Sense Disambiguation based on a loose identification of candidate meanings coupled with a densest subgraph heuristic which selects high-coherence semantic interpretations (Moro, Reganato & Navigli, 2014). The Babelfy parameters that we tuned according to our preferences are:

— *setAnnotationType* was set to identify both concepts and named entities,

— *setMatchingType* was set to exact matching,

— *setMultiTokenExpression* was on to identify multi-word tokens,

— *setScoredCandidates* was set in a way so that it obtains only top-scored candidate from the disambiguation list.

The rest of the parameters were kept to their default value. The named entities identified by both Babelfy and ARK Tagger were allowed to the linking stage. Initially, we provided the original tweet texts as input to Babelfy. We observed that the number of named entities and concepts recognized and linked solely by Babelfy service was quite low. The named entity recognition suffered because of the noisy nature of tweet text. However, the accuracy of the linked resources was satisfactory. So, we modified our system by altering the tweets slightly. We removed the # and considered only the alphabets from an already recognized named entity (tagged by the ARK tagger). After successfully linking such named entities, we searched for more entities which were syntactically similar to the previously known entities. We linked these new entities to corresponding DBpedia resources and also obtained the disambiguation scores.

Clustering of NIL Mentions

The entities which could not be linked to any existing DBpedia resource are supposed to have NIL identifiers so that each NIL may be reused if there are multiple mentions in the text which represent the same (s/similar/identical) entity. We have considered only a spelling based approach here to calculate the similarity between entities. Two unlinked entities are taken to be similar if one of them contains the other (letter only). In that case, the new entity is assigned the same NIL identifier as that of the previous one.

B. Results

We evaluated our approach on the development set consisting of 100 tweets made available by the organizers. In Table 1 we have reported on the official metrics for entity detection, tagging, clustering and linking. The precision, recall and f-scores for the above-mentioned three runs show that the Run 3 produces best results for the task with f-score 0.674, 0.380, 0.252 and 0.646 in the categories Strong Mention Match, Strong Typed Mention Match, Strong Link Match and Mention Ceaf respectively. While all the Runs yield same score in other categories, in Strong Typed Mention Match, we observe better result for our feed-forward neural network model. Our

systems for the three different runs only differ in entity type classification module while all other subtasks follow the same system in all three cases. This results in same result in the last two categories which were mainly the evaluation metrics for linking and nil clustering.

	Precision	**Recall**	**F1**
Run 1			
Strong Mention Match	0.729	0.626	0.674
Strong Typed Mention Match	0.301	0.259	0.278
Strong Link Match	0.586	0.161	0.252
Mention ceaf	0.699	0.600	0.646
Run 2			
Strong Mention Match	0.729	0.626	0.674
Strong Typed Mention Match	0.144	0.124	0.133
Strong Link Match	0.586	0.161	0.252
Mention ceaf	0.699	0.600	0.646
Run 3			
Strong Mention Match	0.729	0.626	0.674
Strong Typed Mention Match	0.411	0.353	0.380
Strong Link Match	0.586	0.161	0.252
Mention ceaf	0.699	0.600	0.646

Table 3.1 Summary of Experimental Results on Development Set

We have developed a hybrid system using the existing Named Entity Recognizer systems and Twitter-specific Part-of-Speech Taggers in conjunction with the classifier developed by us. The Named Entity Linking was done mainly by using Babelfy, which performs as a multilingual encyclopedic dictionary and a semantic network. The performance of our system sued because of certain restrictions in time. The classification module was slightly biased and the accuracy of classification suffered as result of that. Identifying and selecting better features would have improved results. Also a disambiguation module to treat overlapping classes would have been useful. The accuracy of the linking would also improve by taking a semantic similarity approach

using synonym sets for the mentions or context word overlapping from the sets while NIL clustering.

3.4 Challenges of Cross-Genre Entity Identification

The primary focus of our work is on building a system that will be able to automatically acquire knowledge from unstructured texts in different domains or genres. The complexity of this task is twofold: first, the domain is not known beforehand. In most of the ontology building approaches, we generally proceed keeping a particular domain like *sports* or *automobile engineering* in focus. On the contrary, we do not have a fixed domain to start our work with. The datasets we are looking at are just unstructured texts that can be on any random theme or domain. The second challenge is that the datasets are from different genre of text. Quite understandably, texts from a movie review or a blog would not adhere to the same syntactical structure as of a literary work. Tweets and Mythological texts nowhere share exactly same characteristics. So it can be safely assumed that our effort will prove not to be effective if we try to look for entities based on some linguistic patterns as they would vary immensely from domain to domain. So what we try here is to find out the common attributes that these texts possess and use them to build a model to extract appropriate entities, making use of the information these attributes carry.

We began our work with two hypotheses: initially, we planned to view this as a typical domain adaptation problem where we do not have enough or any training data for a particular domain and so we train from some other domain for which we already have enough data points and then apply the model to that domain. As per the plan, we took one dataset at a time for training and then test data from all others, including it, and took note of the system's performance. For the second hypotheses, we thought of a situation where we have data points from both the domains but they are not separated. And we need to build such a model which will be able to yield satisfactory results when this happens. Hence we took all the two-dataset combinations as possible and tried to observe their performance when applied to the test set of same mixed domains as well as separate ones.

As we didn't have gold-standard annotation for other domains except Twitter and Wikipedia Articles, we used Stanford NER tagger to annotate the documents and considered those as gold standard for training and evaluation purpose. We did a 70%-30% division of all the datasets for

training and testing purpose respectively. As the domain is not fixed for all the datasets; even a single dataset can hold multiple topics; we could not come up with domain specific concepts to classify our entities into. Hence, only a few top level classes were considered as entity types. Twitter dataset is annotated with the tags PERSON, LOCATION, ORGANIZATION, THING and EVENT, whereas all the other ones are tagged with PERSON, LOCATION and ORGANIZATION. Moreover, as section 2.7 discusses, the ratio of entity vs. non-entity is not fixed for all the datasets. And as an obvious corollary, we can understand that the entity co-occurrence statistics won't be same for all the datasets either. So it is quite a challenge to train our model using a particular dataset and then evaluate it after testing on another. We discovered that there could be two ways to implement the idea; for the first phase, we used training data from a dataset to build a classifier and then evaluate it against 6 separate test sets, i.e. test data from each domain. This is to study if we can address the problem of not having training data from a particular domain with annotated data from different domain. As we are extracting only the top-level named entities; and that is mostly Person, Location or Organization names, their characteristics do not differ much from dataset to dataset, though the context they appear in changes significantly. For the second phase of our experiment, we combined training data from two datasets and then evaluated that particular model against data from each of those datasets separately, plus two of them combined. This was done to address the problem of having few, but not enough annotated data from the dataset one needs to apply their NER system on.

3.5 Feature Extraction Module

Previous empirical results showed that latent generalizable features can increase the accuracy for out-of-domain prediction performance (Blitzer et al., 2006). It has also been justified by a recent theoretic study that a proper feature representation is crucial to domain adaptation due to its contribution on bridging domain divergence (Ben-David et al., 2007; 2010).

We have incorporated various orthographic, as well as contextual features in our approach to recognize named entities from unstructured texts. As we have several datasets that vary enormously from one genre to other, both in terms of content and inscription style. We have experimented with various features, some of which are valid for all the domains, while some are helpful just for some specific domains. For example, features like if the word is a hashtag will

only work for tweets; for the other datasets, it is just an added overhead. So we have separated our features into two distinct categories: Domain-Independent and Domain-Dependent features.

3.5.1 Domain Independent Features:

A. **POS (current, previous, next):** All the texts were POS tagged using Stanford CoreNLP Parts-of-Speech tagger during the preprocessing phase. We use three features, i.e. POS tag of current word, previous word and next word as features to identify whether or not current word is a named entity.

   For example, "*Never did the Princess leave the palace*." – if '*Princess'* is our current word here, we will consider the POS of '*Princess'*, i.e. Noun, and also the POS of '*the'* and '*leave'* as feature values.

B. **N-gram (bigram, trigram)(previous, next):** In the fields of computational linguistics, an n-gram is a contiguous sequence of n items from a given sequence of text or speech. The items can be phonemes, syllables, letters, words or base pairs according to the application. For this specific task, we consider words as grams. An n-gram of size 1 is referred to as a "unigram"; size 2 is a "bigram"; size 3 is a "trigram". Larger sizes are sometimes referred to by the value of n, e.g., "four-gram", "five-gram", and so on. We only took unigram, bigram and trigram of current work in both the directions; i.e. previous and next bigram and trigram.

   For example, "*Never did the Princess leave the palace*." – If '*Princess'* is our current word here, we will consider the word '*Princess'* as unigram, '*the Princess'* and '*Princess leave'* as both way bigrams, and '*did the Princess'*, '*Princess leave the'* as the trigram feature values. So, we have considered all the words in a context window span of 5 words, with current word at middle.

C. **isAllCaps:** This Boolean feature was kept to catch the abbreviated named entities like USA, or IBM etc. It checks if all the characters of present word are in capital letters.

D. **isAllLower:** Contrary to the previous one, this feature checks if all the characters of current word are in lowercase. In such cases, the probability of the current word being a non-entity is high. This is also a Boolean feature like the previous one.

E. **isMixedCase:** This feature returns true if the current word is a combination of uppercase and lowercase characters.
For example: 'Peru' returns true, while 'USA' or 'x-mas' returns false.

F. **isInitCaps:** This feature notes in only the initial letter of current word is capitalized. We can consider this as domain independent because for all the domains, it is quite a regular practice for named entities to have their 0^{th} position character in noun.
For example: 'Alabama' returns true, while 'house' returns false.

G. **containsRoman:** This feature needs to be added because a few of the PERSON-typed named entities used by authors contain Roman symbols and/or digits.
For example, *"His elder son Robert Williams II was declared the next king to rule over northern regions"*: In this sentence, we need to identify all these words "Robert Williams II" as a PERSON type entity.

H. **isFunctionWord:** This feature holds true if the POS tag of Current word is preposition or article.

I. **isSingleChar:** This feature returns true if current word contains only a single character. Most of the times, named entities are more than a character long; however, in some cases, a single character can appear as an individual entity as well.
For example: "*The R Company is currently the biggest threat to this nation*.": In this sentence, the letter R is a part of an ORGANIZATION-type named entity- 'The R Company'

J. **isPrevWordArticleorNoun:** To emphasize on the previous word's POS, we check if the previous word is an article or if it is a noun. Researches show that in these cases, these two features can play an important role identifying the possible entities.

K. **wordLength:** We keep the length of current word as a feature.
For example, "Never did the Princess leave this palace." If 'Princess' is the current word, this feature will return the value 8.

L. **termFrequency:** This is basically the count of how many times a particular term/word has appeared in the total dataset. As the datasets vary in length, to keep this feature value unbiased, we normalize it by dividing with the total word count of the dataset.

3.5.2 Domain Dependent Features:

These domain dependent feature sets are used along with the ones discussed above to identify the named entities in tweets, blogs, reviews or other used generated web contents. It is a common practice among twitter or other social media users to use symbols or special characters like hashtags while generating content. To increase the recall of our system for these datasets, we have to include these features to our classifier. However, feature overfitting can lead to performance degradation in other domains.

A. **containsDot:** This Boolean feature returns true if the word we are considering contains a dot. Usage of 'dot' is pretty common in case of account names or mail ids of social media users.
For example, "*Ra.One turned out to be an utter disappointment*" – In this sentence, the word '*Ra.One'* will return true for this feature value.

B. **containsHyphen:** Similar to the previous one, this feature returns true if the word we are considering contains a hyphen.
For example, the term '*PhotoshopCS-4'* returns true.

C. **isURL:** Presence of URLs is common in data generated for web, and hence, to address these values, we have introduced this feature which returns true if the current token is an URL.

D. **isInitSymbol:** If the first, i.e. 0^{th} character of current word is a symbol, this feature returns true. We can call this a special feature for domains like twitter where usage of '*@*' is common for profile names and '#' for trending topics.
For example, this feature returns true for the token *'@gooner_rafa'* or *'#PotterMania'*

E. **containsDigit:** This feature returns true if the current token contains digits in any position of the word.
For example, the terms '*Bond007'* and '*Oct31_Halloween'* returns true.

3.6 Classification Module

After the feature extraction and feature file building phase, we sent our training files to a CRF based classifier. CRF or Conditional Random Fields are a class of statistical modeling method often applied in pattern recognition and machine learning, where they are used for structured prediction. Whereas an ordinary classifier predicts a label for a single sample without regard to "neighboring" samples, a CRF can take context into account. It is often used for labeling or parsing of sequential data, such as natural language text or biological sequences and in computer vision.

We have used CRF++ (version 0.58)[4], a simple, customizable, and open source implementation of Conditional Random Fields (CRFs) for segmenting/labeling sequential data. It is designed for generic purpose and can be applied to a variety of NLP tasks, such as Named Entity Recognition, Information Extraction and Text Chunking.

We have evaluated our system for Domain Independent feature set once and then including all the features. However, the evaluation results did not appear to be satisfactory. Hence, an additional *'Tag Rectifier'* module is introduced in the system after the classifier.

[4] https://taku910.github.io/crfpp/#download

3.7 Tag Rectifier Module

The output from the classifier is fed to a noise cleansing module which works in two phases.

First, it checks whether a stopword or a non-noun term has been tagged as an entity. If yes, our system negates the decision made by classifier and tags it as non-entity. This is done to increase the precision of system by avoiding non-entities being tagged as entities.

In the second phase, a checking is done for all the noun words against our gazetteer entries. The role of the gazetteer is to identify entity names in the text based on lists of already known names of specific entity types. We have used the ANNIE Gazetteer lists that come with the GATE[5], which is open source software used for text processing. GATE was originally developed in the context of Information Extraction (IE) R&D, and IE systems in many languages and shapes and sizes have been created using GATE with the IE components that have been distributed with it. GATE is distributed with an IE system called ANNIE, A Nearly-New IE system (developed by Hamish Cunningham, Valentin Tablan, Diana Maynard, Kalina Bontcheva, Marin Dimitrov and others). The gazetteer lists used are plain text files, with one entry per line. Each list represents a set of names, such as names of cities, organizations, days of the week, etc. Though there are default methods available in ANNIE for gazetteer matching, we developed our own matching algorithm against our own gazetteer list for each of the entity type by merging some of the lists from ANNIE.

For each word with a noun POS tag, we do the following:

i. Check whether the word matches with names from any of the lists.
ii. If no match is found, keep the classifier outcome as final and go to step v. Otherwise go to step iii.
iii. If the word matches with only one list (i.e. a single entity type) and the word starts with a capitalized character, tag that particular word with the corresponding entity type of the list it was matched against, and go to step v. Otherwise go to step iv.

[5] https://gate.ac.uk/

iv. When the system finds match from multiple lists (i.e. more than one entity type) for a single word, it relies on the classifier used in previous step of entity extraction for entity type disambiguation and keeps the tag given by the classifier as correct tag.

v. End.

3.8 Results and Observations

Though the amount of digitalized texts are rapidly increasing; as a language varies so widely, collecting and curating training sets for each different domain is prohibitively expensive. At the same time, the differences in vocabulary and writing style across domains can cause the state-of-the-art supervised models to dramatically increase the error. Domain adaptation methods provide a way to alleviate such problem of creating training sets for different domains by generalizing models from a resource-rich source domain to a different, resource-poor target domain. We have considered two scenarios in this context, which are descried next.

3.8.1 Hypothesis 1: Single Domain Training

For our first hypothesis, we considered that tagged data is available only from one domain and using that as training set, we need to build a system that'll work for all the other domains. Hence, we built seven different classifiers using seven different datasets as training separately. The datasets were split in a 70%-30% ratio for training and testing purpose respectively. Then using each of these classifier models one by one, test sets for all the datasets were tagged and evaluated.

Initially we built our feature files for all the datasets including both domain dependent and domain independent features and fed them to a CRF-based classifier. The outcomes for each dataset were evaluated in terms of Precision, Recall, F-Measure and Accuracy.

***Precision*:**

This is the percentage of correctly identified entities among all the entities detected. The formula to measure precision is: *{TP/(TP+FP)}*100%*

***Recall*:**

This is the percentage of entities detected among all the entities present in the dataset. The formula to measure recall is: *{TP/(TP+FN)}*100%*

F-Measure:

The traditional F-measure or balanced F-score (F1 score) is the harmonic mean of precision and recall, and the formula is: *(2*Precision*Recall)/(Precision + Recall)*

***Accuracy*:**

Accuracy is used as a statistical measure of how well a binary classification test correctly identifies or excludes a condition. That is, the accuracy is the proportion of true tags (correctly tagged as entity or correctly tagged as non-entity) among the total number of words tagged. The formula to measure accuracy is: *{(TP+TN)/(TP+TN+FP+FN)}*100%*

For all the metrics considered,

TP (True Positive)= number of entities that are correctly extracted;
TN (True Negative)= number of non-entities that are correctly rejected;
FP (False Positive)= number of non-entities that were extracted as entities;
FN (False Negative)= number of entities that were rejected as non-entities.

Train Set	Test Set Results(%)				
	Domain	Precision	Recall	F-measure	Accuracy
Twitter	Twitter	54.13	30.80	39.26	89.37
	Wikipedia	36.81	14.45	20.75	87.64
	Review	46.51	8.69	14.65	98.16
	Blog	48.04	19.21	27.45	97.89
	NEWS	35.38	1.68	3.21	92.31

	Literature	13.48	11.72	12.54	88.53
	Mythology	10.01	21.75	13.71	77.01
Wikipedia Article	Twitter	33.01	28.57	30.63	85.56
	Wikipedia	87.54	85.32	86.41	96.99
	Review	34.01	43.47	38.16	96.43
	Blog	44.65	86.67	58.93	97.49
	NEWS	59.34	15.82	24.98	92.79
	Literature	29.25	23.75	26.22	90.62
	Mythology	42.13	77.59	54.60	90.52
Review	Twitter	22.22	18.06	19.58	88.79
	Wikipedia	33.33	9.20	18.35	88.79
	Review	85.00	29.56	43.87	98.81
	Blog	-	-	-	97.88
	NEWS	51.78	2.12	4.08	92.42
	Literature	84.61	8.42	16.69	93.03
	Mythology	-	-	-	92.00
Blog	Twitter	46.84	8.94	15.01	88.71
	Wikipedia	89.51	41.23	56.46	92.87
	Review	40.00	8.69	17.02	98.42
	Blog	73.33	47.45	57.62	98.55

	NEWS	90.25	15.60	26.60	93.47
	Literature	84.45	12.49	21.76	93.70
	Mythology	73.28	51.35	60.38	95.05
NEWS	Twitter	37.14	16.44	22.79	87.57
	Wikipedia	78.17	62.31	69.34	93.83
	Review	5.40	0.86	1.49	98.20
	Blog	66.28	68.62	67.43	98.62
	NEWS	84.71	85.27	84.99	97.71
	Literature	19.49	13.71	16.10	89.97
	Mythology	23.01	54.18	32.30	83.31
Contemporary Literature	Twitter	44.86	13.82	21.13	88.49
	Wikipedia	87.80	50.02	63.73	93.62
	Review	40.00	8.69	17.02	98.42
	Blog	63.93	45.88	53.42	98.33
	NEWS	81.18	28.13	41.78	94.05
	Literature	83.90	21.18	39.74	93.62
	Mythology	77.24	47.65	58.94	95.12
Mythology	Twitter	44.41	23.09	30.38	88.19
	Wikipedia	82.82	75.24	78.85	95.48
	Review	100	43.47	86.58	98.43

	Blog	59.51	77.25	67.23	98.43
	NEWS	87.35	5.56	10.46	92.77
	Literature	55.70	13.10	21.21	93.17
	Mythology	86.13	91.29	88.64	98.28

Table 3.2 Evaluation Results for Entity Identifier Including All Features

The instances where the training and testing data are from same domain are highlighted. As we can see, models built from Wikipedia Article data, NEWS data and Mythology data performs very well for test sets from same domain, while the Review model performs poorly for all the datasets. It fails to detect even a single entity in two domains. Also, all the models perform poorly in terms of recall against the review test set, which means the classifiers are not able to detect enough number of valid entities from the review texts. We noticed that the recall for twitter test set is maximum for the model which was built using twitter as training set.

As there are some features which works well only for specific domain and we are trying to build a system that can yield good amount of Named Entities from any genre of text given, we tried to note its performance based on only the domain independent feature set. Though the evaluation scores reduced for a few particular instances, on a whole, the differences among different entity sets were more balanced.

Train Set	Test Set Results(%)				
	Domain	Precision	Recall	F-measure	Accuracy
Twitter	Twitter	53.72	30.39	38.82	89.32
	Wikipedia	37.39	14.54	20.94	87.70
	Review	42.55	8.69	14.44	98.13
	Blog	48.97	18.82	27.19	97.90

	NEWS	34.92	1.61	3.08	92.00
	Literature	12.40	12.32	12.36	88.80
	Mythology	7.05	15.16	9.62	77.08
Wikipedia Article	Twitter	32.62	28.58	30.47	85.45
	Wikipedia	87.45	85.36	86.39	96.98
	Review	32.78	43.47	37.38	96.36
	Blog	43.93	86.66	58.31	97.42
	NEWS	61.61	18.46	28.41	92.66
	Literature	26.81	25.90	26.35	90.71
	Mythology	41.01	77.58	53.65	90.15
Review	Twitter	25.00	12.04	16.25	88.82
	Wikipedia	-	-	-	88.78
	Review	85.00	29.56	43.87	98.81
	Blog	-	-	-	97.89
	NEWS	54.83	1.24	2.43	92.13
	Literature	100	16.87	33.69	93.60
	Mythology	-	-	-	92.04
Blog	Twitter	46.62	8.94	15.01	88.70
	Wikipedia	89.58	41.55	56.77	92.91
	Review	40.00	8.69	17.02	98.42

	Blog	73.00	46.66	56.93	98.53
	NEWS	88.53	10.18	18.26	92.81
	Literature	85.29	12.23	21.40	94.23
	Mythology	73.23	51.21	60.27	95.04
NEWS	Twitter	38.69	17.02	23.64	87.74
	Wikipedia	76.62	52.04	61.98	92.85
	Review	42.53	8.69	14.44	98.13
	Blog	64.50	58.43	61.31	98.46
	NEWS	84.81	85.93	85.37	97.67
	Literature	19.95	13.33	15.98	91.01
	Mythology	18.53	36.32	24.54	83.59
Contemporary Literature	Twitter	44.90	13.67	20.96	88.50
	Wikipedia	87.97	49.47	63.32	93.58
	Review	40.00	8.69	17.02	98.42
	Blog	63.73	45.49	53.08	98.33
	NEWS	81.37	23.36	36.31	93.53
	Literature	82.91	41.05	54.91	94.15
	Mythology	79.03	46.20	58.31	95.14
Mythology	Twitter	44.17	22.71	29.99	88.18
	Wikipedia	82.74	75.05	78.71	95.45

	Review	100	43.47	86.58	98.43
	Blog	59.75	76.86	67.23	98.44
	NEWS	85.29	6.37	11.86	92.53
	Literature	55.23	14.68	23.20	93.76
	Mythology	84.36	91.03	87.57	98.10

Table 3.3 Evaluation Results for Entity Identifier Including Only Domain Independent Features

As it can be seen, the scores for a few in-domain (training and testing data are from same domain) classifiers have deteriorated slightly (at most 1%), but the difference among the results for different sets of training data for a single classifier model has reduced significantly.

The tags which are considered for Entity Identification and Classification task are: PERSON, LOCATION, ORGANIZATION for all the datasets and two extra tags THING and EVENT for twitter data only. All other words except these were tagged as non-entity or OTHERS initially. So this particular class 'OTHERS' had the lion's share of words from all the datasets, which eventually made our system biased towards this particular class. As a result, the system showed visible decline in recall score for all the domains as it had the tendency to tag most of the entities as OTHERS or non-entity. To remove this biasness towards non-entities, we divided the class OTHERS into Parts-Of-Speech specific classes. This essentially means that an adverb which is non-entity will be assigned the tag OTHERS-ADVERB, while a noun which is not an entity will go to the class OTHERS-NOUN. This strategy helped to attain a greater balance in terms of words per class ratio and thus removed the bias that caused poor recall. We followed this strategy in both of our hypothesis.

As the results for some of the cross-domain (test and train data from different domain) classification are not satisfactory, we introduced a final noise clearance module using the gazetteer list prepared on the outcomes from domain-independent feature set. As discussed earlier, this module works in two phases improving precision and recall of entity identification respectively. The scores after applying this module are listed below:

Train Set	Test Set Results(%)				
	Domain	Precision	Recall	F-measure	Accuracy
Twitter	Twitter	75.62	69.13	72.23	94.07
	Wikipedia	76.97	23.38	35.86	90.63
	Review	23.78	14.78	18.23	97.91
	Blog	39.47	23.53	29.48	97.66
	NEWS	38.61	6.95	11.79	91.79
	Literature	88.76	68.01	77.02	97.39
	Mythology	35.36	8.44	13.62	92.14
Wikipedia Article	Twitter	61.06	62.13	61.59	91.61
	Wikipedia	85.57	93.88	89.53	97.54
	Review	21.38	14.78	17.48	97.80
	Blog	49.78	88.23	63.65	97.90
	NEWS	64.67	24.54	35.58	92.99
	Literature	77.12	84.47	80.63	97.40
	Mythology	64.62	79.43	71.26	95.27
Review	Twitter	99.25	35.01	51.76	93.09
	Wikipedia	98.48	7.87	14.57	89.66
	Review	88.77	37.83	53.04	98.95
	Blog	66.66	7.84	15.50	97.93

	NEWS	81.96	4.35	8.26	92.37
	Literature	100	69.68	82.13	97.71
	Mythology	34.86	4.87	8.55	92.40
Blog	Twitter	76.83	45.96	57.51	92.85
	Wikipedia	91.37	47.81	62.77	93.11
	Review	38.03	12.98	19.35	98.20
	Blog	61.62	65.49	63.49	98.43
	NEWS	74.88	16.31	26.78	92.92
	Literature	95.87	77.38	85.64	97.73
	Mythology	68.07	53.59	59.97	94.94
NEWS	Twitter	70.55	53.30	60.72	92.58
	Wikipedia	82.00	58.75	68.45	94.10
	Review	17.31	13.47	15.15	97.63
	Blog	53.97	61.17	57.35	98.11
	NEWS	76.35	94.13	84.31	97.23
	Literature	84.38	78.57	81.37	97.62
	Mythology	63.03	43.65	51.58	93.88
Contemporary Literature	Twitter	73.16	49.45	59.01	92.64
	Wikipedia	88.21	56.28	68.71	94.36
	Review	50.94	11.74	19.08	98.43

	Blog	61.54	47.06	53.33	98.28
	NEWS	78.24	27.39	40.58	93.67
	Literature	89.34	79.24	83.99	98.06
	Mythology	76.33	51.09	61.21	95.22
Mythology	Twitter	66.61	59.12	62.64	92.32
	Wikipedia	83.00	82.24	82.62	96.12
	Review	40.32	10.86	17.12	98.34
	Blog	57.06	77.65	65.78	98.32
	NEWS	68.14	11.28	19.35	92.58
	Literature	87.22	80.08	83.50	97.97
	Mythology	83.59	93.34	88.19	98.16

Table 3.4 Evaluation Results for Entity Identifier Including Tag Rectifier Module

Significant improvement in terms of both precision and recall was observed after including this final module to our system. While the in-domain f-measure for the domains Mythology, Wikipedia Article, NEWS and Literature are all above 80%, the results for the user generated contents, i.e. Blog, Review and Twitter did not cross the 80% mark. Though, it can be safely said that the results have improved than the original outcome before implementing this module.

Another interesting observation that we could make is that the accuracy of our system is almost always above 90%. This happens due to the large number of True Negatives; i.e. our system correctly detected a large number of non-entities.

3.8.2 Hypothesis 2: Mixed Domain Training

In the second hypothesis, we trained our model using data from combination of two domains, and then the model was used to tag data from those domains separately as well as mixed. For training, we have taken 20,000 instances from each domain, i.e. a total of 40,000 instances for training and for testing we have taken 5,000 from each and merged them to have a test set of 10,000 instances. However, for testing using data from single domain, we kept the same test set of Hypothesis 1. The results are evaluated using same metrics that we used for Hypothesis 1, i.e. Precision, Recall, F-measure and Accuracy, and the scores are listed as follows:

Training	Testing				
	Domain	**Precision**	**Recall**	**F-Score**	**Accuracy**
News + Blog	News	75.52	92.01	82.96	97.02
	Blog	61.46	72.54	66.54	98.48
	Mixed	83.85	88.87	86.29	98.60
News + Wikipedia	News	75.51	91.50	82.74	96.99
	Wikipedia	85.89	91.94	88.81	97.40
	Mixed	89.78	88.75	89.26	97.84
News + Twitter	News	75.26	92.08	82.83	96.99
	Twitter	75.01	64.36	69.28	93.63
	Mixed	87.44	71.65	78.76	96.50
News + Review	News	75.58	91.86	82.93	97.02
	Review	83.33	30.43	44.58	98.81
	Mixed	83.76	81.02	82.37	98.33

News + Literature	News	75.66	91.79	82.95	97.02
	Literature	87.62	80.08	83.68	97.99
	Mixed	90.47	84.00	87.12	97.61
News + Mahabharat	News	75.65	92.45	83.21	97.05
	Mahabharat	82.97	91.23	86.90	97.98
	Mixed	83.34	91.01	87.01	98.03
Blog + Literature	Blog	61.40	68.63	64.81	98.45
	Literature	89.35	77.89	83.22	97.98
	Mixed	93.01	79.55	85.76	97.07
Blog + Mythology	Blog	60.70	74.51	66.90	98.46
	Mythology	83.46	91.82	87.44	98.06
	Mixed	83.07	92.94	87.73	98.33
Blog + Wikipedia	Blog	59.80	72.94	65.72	98.42
	Wikipedia	86.71	81.04	83.78	96.48
	Mixed	91.71	82.79	87.02	96.91
Blog + Review	Blog	60.56	67.45	63.82	98.41
	Review	87.87	25.21	39.18	98.77
	Mixed	86.67	26.71	40.84	98.81
Blog + Twitter	Blog	60.0	49.41	54.19	98.26
	Twitter	75.0	59.90	66.61	93.30

	Mixed	73.36	59.67	65.81	93.20
Literature + Mythology	Literature	88.35	80.67	84.34	98.08
	Mythology	83.97	90.51	87.12	98.03
	Mixed	80.08	86.31	83.08	98.28
Literature + Wikipedia	Literature	88.70	79.49	83.84	98.03
	Wikipedia	86.86	71.51	78.44	95.59
	Mixed	89.79	73.82	81.03	97.18
Literature + Review	Literature	89.41	76.28	82.33	97.90
	Review	89.70	26.52	40.93	98.79
	Mixed	74.54	43.39	54.85	98.60
Literature + Twitter	Literature	90.45	71.98	80.17	97.71
	Twitter	74.62	59.09	65.96	93.19
	Mixed	74.11	58.96	65.67	95.89
Mythology + Wikipedia	Mythology	84.92	88.72	86.78	98.01
	Wikipedia	86.33	80.85	83.50	96.42
	Mixed	88.15	84.56	86.32	97.32
Mythology + Review	Mythology	81.40	93.21	86.90	97.93
	Review	95.45	18.26	30.65	98.70
	Mixed	83.78	77.26	80.39	98.13
Mythology + Twitter	Mythology	85.86	85.69	85.78	97.91

	Twitter	73.49	59.45	65.73	93.08
	Mixed	79.49	71.81	75.45	95.74
Wikipedia + Review	Wikipedia	85.51	90.19	87.79	97.19
	Review	96.55	24.34	38.88	98.79
	Mixed	81.65	81.09	81.37	97.73
Wikipedia + Twitter	Wikipedia	86.35	75.14	80.36	95.88
	Twitter	72.03	66.17	68.98	93.36
	Mixed	77.15	72.42	74.71	94.97
Review + Twitter	Review	95.16	25.65	40.41	98.81
	Twitter	74.80	59.72	66.42	93.26
	Mixed	73.42	58.69	65.23	93.81

Table 3.5 Evaluation Results for Entity Identifier Considering Hypothesis 2

In contrast to results obtained in Hypothesis 1, we can observe that all the results have improved if we have our classifier model trained with data from both the domains, instead of a single domain. Except Review, our system achieved satisfactory result with above 80% F-measure in many test cases. Both precision and recall have improved significantly for the user generated web contents like Twitter and Blog entries. Similar to Hypothesis 1, our system delivered excellent results in terms of Accuracy. We can conclude that our system performs better if we have at least a few training instances from the domain we have as test set.

CHAPTER 4

TAXONOMIC RELATION EXTRACTION

Ontologies are carefully designed to cultivate the domain at hand and along with the entities detected, we also need to have an understanding of how they are related. Hence, this field of relation extraction came into picture with gigantic research scope. There can be many factors associated with understanding the meaning of an unstructured text. The difficulty lies both in identifying those factors and then coming up with algorithms and implementing methods to handle those factors effectively in order to achieve a consensus upon retrieved relations among entities. This process can either be supervised or unsupervised or hybrid.

Supervised: The cases where we know beforehand which exact relations we need to extract are called supervised relation extraction. Generally in a domain which has been explored before, we already know the type of relations possible for the entities of that domain and so it is easier to follow that guideline and try to find more instances of that relation.

Unsupervised: For the domains which have not been previously explored before, we need to find methods first to identify the type of relations possible in that domain before actually trying to extract few instances.

Hybrid: While extending a previously extracted ontology, along with populating the existing relation structures with more number of instances, we may also try to search and check if some different types of relations can exist for that domain.

Relations, on the other hand, can be of two major types, taxonomic and non-taxonomic relations. To describe them briefly, taxonomic relations are the ones which have a hierarchical or tree structure. On the contrary, non-taxonomic relations do not display such natural structures, they are atomic in nature and quite difficult to discover than taxonomic ones.

4.1 Introduction

Taxonomies are useful for building and maintaining different aspects of knowledge, most of which can be mathematically expressed with partial orders. These kinds of relations are used for

representing information at appropriate levels of generality and automatically reducing it to more domain specific concepts by means of a mechanism of inheritance (Woods, 1991). Taxonomy or hierarchical relations between ontological concepts are considered as useful tools for content organization, navigation, and retrieval (domain ontologies), as well as to provide valuable input for semantically intensive tasks such as question answering and textual entailment (application ontologies).

The ability of systems to acquire desired knowledge from taxonomies depends on the definition, identification and organization of taxonomic information (Fall, 1996). Taxonomies can be examined from three different perspectives: structurally, ontologically and semantically. From a structural perspective, the way knowledge is represented does not always follow an appropriate level of clarity for computers to reason from it. Additional structural constraints have been suggested in order to make taxonomies more usable in application contexts. One such constraint is that two sibling categories be incompatible. For example, the concepts "physical state" and "mental state" are children of "state" and incompatible. Both concepts are incompatible in the ontology because the former involves a physical object whereas the latter involves a mental object (Bouaud et al., 1994). From the perspective of formal ontology, Guarino gives several examples of is-a overloading. For example, "a physical object is an amount of matter" reflects a reduction of sense, since a physical object is more than just an amount of matter (Guarino, 1999). Guarino & Welty focus on meta-properties that help formalize constraints on the taxonomic relation. From the standpoint of semantics, Brachman describes several meanings of the is-a relation that may exist between two generic concepts in semantic networks (subset/superset, generalization/specialization, kind-of, conceptual containment, role value restriction, set/prototype) (Brachman, 1983). He also suggests using those semantic subcomponents as the primitives of a representation system. In practice, taxonomic knowledge is complex and remains partially intuitive in many existing ontologies. This may lead to ruptures in knowledge representation, and thus impair the capability of reasoning from the system.

4.2 Related Work

There are several works that aim at building taxonomies and ontologies which organize concepts and their taxonomic relations into hierarchical structures. Snow et al. constructed classifiers to identify hypernym relationship between terms from dependency trees of large corpora (Snow et

al., 2005; Snow et al., 2006). Terms with recognized hypernym relation are extracted and incorporated into a man-made lexical database, WordNet (Fellbaum, 1998), resulting in the extended WordNet, which has been augmented with over 400, 000 synsets. (Ponzetto and Strube, 2007) and (Suchanek et al., 2007) both mined Wikipedia to construct hierarchical structures of concepts and relations. While the former exploited Wikipedia category system as a conceptual network and extracted a taxonomy consisting of subsumption relations, the latter presented the YAGO ontology, which was automatically constructed by mining and combining Wikipedia and WordNet. The idea of using lexico-syntactic patterns in the form of regular expressions for the extraction of semantic relations, in particular taxonomic relations has been introduced by Hearst (Hearst, 1992). Pattern-based approaches in general are heuristic methods using regular expressions that originally have been successfully applied in the area of information extraction (Hobbs, 1993). In this lexico-syntactic ontology learning approach the text is scanned for instances of distinguished lexico-syntactic patterns that indicate a relation of interest, e.g. the taxonomic relation. Thus, the underlying idea is very simple: Define a regular expression that captures re-occurring expressions and map the results of the matching expression to a semantic structure, such as taxonomic relations between concepts.

For example, the following provides a sample pattern-based ontology extraction scenario. In the paper by Hearst, the following lexico-syntactic pattern was considered:

…NP{, NP}{,} or other NP…*

When we apply this pattern to a sentence it can be inferred that the NP's referring to concepts on the left of or other are sub concepts of the NP referring to a concept on the right. For example from the sentence

Bruises, wounds, broken bones or other injuries are common.

System can extract the taxonomic relations (BRUISE,INJURY), (WOUND,INJURY), and, (BROKEN–BONE,INJURY).

Hearst defined the patterns manually, which is a time-consuming and error-prone task. In a later work (Morin, 1999), the work proposed by Hearst is extended by using a symbolic machine learning tool to refine lexico-syntactic patterns. In this context the PROMETHEE system has been presented that supports the semi-automatic acquisition of semantic relations and the

refinement of lexico-syntactic patterns. The work of Assadi (Assadi, 1999) reports a practical experiment of construction of a regional ontology in the field of electric network planning. He describes a clustering approach that combines linguistic and conceptual criteria. As an example he gives the pattern <NP, line> which results in two categorizations by modifiers.

(Faure & Nedellec, 1998) have presented a cooperative machine learning system called ASIUM which is able to acquire taxonomic relations from syntactic parsing. The ASIUM system is based on a conceptual clustering algorithm. Basic clusters are formed on head words that occur with the same verb after the same preposition. ASIUM successively aggregates clusters to form new concepts and the hierarchies of concepts form the ontology. The ASIUM approach differs from the approach in this work because the relation learning is restricted to taxonomic relations. An ontology learning system where the different techniques have been applied on dictionary definitions in the context of the insurance and telecommunication domains is described in two papers (Maedche & Staab, 2000) (Kietz et al., 2000). An important aspect in this system and approach is that existing concepts are included in the overall process. Thus, in contrast to Hearst and Morin, the extraction operations have been performed on the concept level, thus, patterns have been directly matched onto concepts. Thus, the system is, beside extracting taxonomic relations from scratch, able to refine existing relations and refer to existing concepts.

On the other hand, information extraction bootstrapping algorithms automatically harvest related terms on large corpora by starting with a few seeds of pre-specified relations (e.g. is-a, part-of) (Pantel and Pennacchiotti, 2006; Kozareva et al., 2008). Bootstrapping algorithms rely on some scoring function to assess the quality of terms and additional patterns extracted during bootstrapping iterations. Similarly, but with a different focus, Open IE, (Banko and Etzioni, 2008; Davidov and Rappoport, 2008), deals with a large number of relations which are not pre-specified. Either way, the output of these algorithms is usually limited to a small number of high-quality terms while sacrificing coverage (or vice versa). Recently, (Baroni and Lenci, 2010) described a general framework of distributional semantic models that extracts significant contexts of given terms from large corpora. Consequently, a term can be represented by a vector of contexts in which it frequently appears. Any vector space model could then use the terms' vectors to cluster terms into categories. Sibling terms (e.g. Honda, Toyota), therefore, have very high chance to be clustered together. Nevertheless, this approach cannot recognize ancestor

relations. In a later work, researchers compare TAREC with this framework only on recognizing sibling vs. no relation, in a strict experimental setting which pre-specifies the categories to which the terms belong (Do & Roth, 2000).

4.3 SemEval-2016 Task 13: Taxonomy Extraction & Evaluation

This subsection describes our approach to build a language-independent hypernym extraction system, based on two modules for the SemEval-2016 Task 13 on Taxonomy Extraction Evaluation (TExEval-2). This task focuses only on the hypernym-hyponym relation extraction from a list of terms collected from various domains and languages. The first module of our system is built on the state-of-the-art system using BabelNet while the second one deals with the parts found within terms and which are useful to establish a hierarchical relation among them. Our system performed well in terms of recall in most of the domains irrespective of the languages; however, the precision scores indicate a scope of improvement. In case of overall ranking, our present system stands fourth in monolingual (i.e. English) evaluation and second in multilingual (i.e. Dutch, Italian, French) setup.

A. Problem Description

SemEval-2016 Task 13: Taxonomy Extraction Evaluation (TExEval-2) has its main focus on hypernym-hyponym relation extraction from given lists of terms collected from multiple domains like Food, Environment and Science (Bordea et al., 2016). This year, the task organizers have extended the problem setup to address the multilingual structure. Along with English, there were terms in French, Dutch and Italian as well for all the domains. For this particular task, we did not have to go through the complexities of entity identification from a text as the lists of terms were already given.

i. One of the main challenges was that we were not provided with any annotated or plaintext corpus that we can use as training. However, the organizers suggested that it would be helpful if we explore the Wikipedia dump for the same.

ii. Second big challenge was to develop a system that will work for languages we don't understand. Ontology development being such a task where some basic domain knowledge is inevitable, this multilingual setup was indeed a great concern for us.

iii. We were specifically asked not to use the resources we most frequently use in this kind of tasks as they were used to construct the gold standard. The list of the resources that were prohibited is:

- hypernym-hyponym relations from the WordNet[6],

- skos:broader and skos:narrower relations from EuroVoc[7],

- the Google product taxonomy[8],

- the Taxonomy of fields and their subfields provided for the National Academies of Sciences, Engineering, and Medicine[9].

However, in contrast, we were free to add more terms if needed to the term lists that were provided by the organizers.

B. System Description

In the present challenge, we had to keep three main points in mind. We wanted to make a single system appropriate for a multilingual setup (Dutch, French, Italian and English). However, it became more difficult as we were not allowed to use any of the widely used resources like WordNet, EuroVoc, Google Product Taxonomy etc. Building a taxonomy which would provide structured information about semantic relations between words is an extremely slow and labor-intensive process. Therefore, we kept our focus on building a system which would be simple and significantly light in terms of computation time. Our system has two main modules, as shown in Figure 4.1:

i. Extracting semantic relations from BabelNet.

ii. Analyzing the terms to find a subterm suitable to become a hypernym.

[6] https://wordnet.princeton.edu/

[7] http://eurovoc.europa.eu/

[8] https://www.google.com/basepages/producttype/taxonomy.en-US.txt

[9] http://sites.nationalacademies.org/PGA/Resdoc/PGA_044522

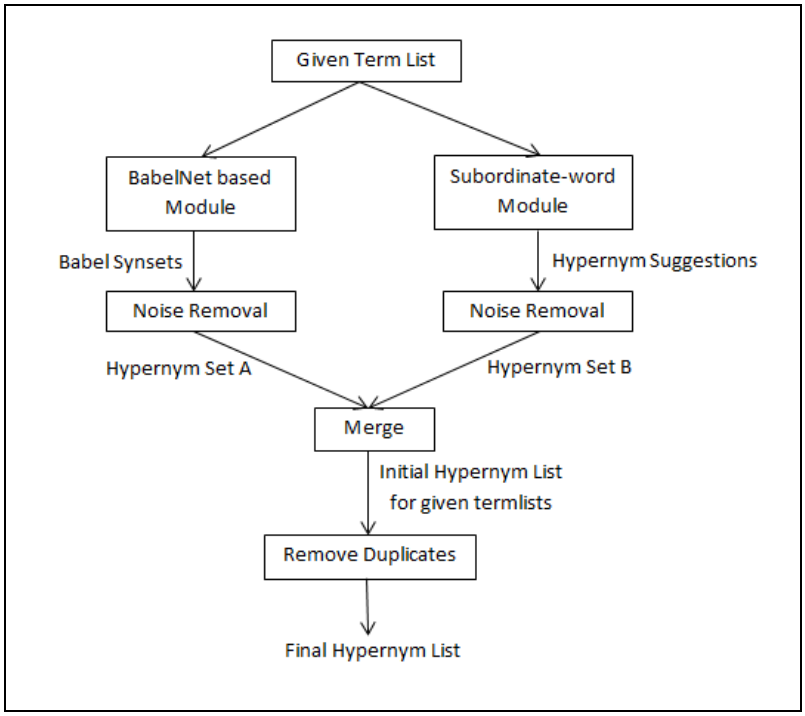

Fig 4.1 Basic system diagram for SemEval Task

B.1 BabelNet Based Module

BabelNet is an open source resource containing both multilingual dictionary with lexicographic and encyclopedic coverage of terms, and a network of concepts and named entities connected in a very large network of semantic relations, called Babel synsets (Navigli and Ponzetto, 2012). Each of the Babel synsets represents a given meaning and contains all the synonyms which express that meaning in a range of different languages. BabelNet 3.5 covers 272 languages, which also include our task related languages like English, French, Dutch and Italian.

Finding out semantic relations from the entire Wikipedia dump with a pattern based approach proved to be quite a long process and computationally expensive as there can be numerous types of valid patterns that can hold a hypernym-hyponym relation. On the other hand, it would take days to initially start with a few patterns and then search for more with a bootstrapping approach. On the other hand, BabelNet already provides a variety of semantic relations for a large number of concepts using knowledge from various resources available including Wikipedia. So, our system execution time gets significantly reduced if we just use the semantic relation set available in BabelNet instead of extending the Wikipedia corpus and analyzing it for the pattern search. Secondly, we wanted to have a system that would fit into the multilingual setup that the task

intends to have this year. The facts were that we do not have a satisfactory amount of knowledge required for identifying the valid patterns for hypernym-hyponym relations in languages other than English, and we also do not have an annotated training data to learn those patterns via a bootstrapping method for those languages. Therefore, it was essential for us to have a tool that could automatically extract such knowledge from corpus. For each term appears in each domain, we obtain a synset from the BabelNet for hypernym relations found over the Wikipedia articles in different languages. We only consider the terms for their NOUN POS tag sense, with the language mentioned in the query. We only considered the NOUN POS tags because it was seen from our observation of term lists, that they contain terms which are mostly nouns. We get the synset for each term which contains a lot of noise such as: repetitive sense words, out-of-domain senses, senses in different morphological form than the existing terms etc. We fed the raw synset output to a cleansing module which would give us only the unique in-domain terms in their correct morphological form as given in the term-list. We further extended this module to find the synsets of the terms present in the cleansed output in order to obtain the entire hypernym tree for the given term which helps to increase the recall of our system.

B.2 Subordinate-word Module

This module deals with finding appropriate parts of given terms that can possibly be the hypernym of the original term. For example, Fruit Custard is a type of Custard. Now these subordinate-words which are potential hypernyms can be of the following two types.

- The subordinate-word can itself be an independent term present in the term-list given. For example, if we have both the terms Biochemistry and Chemistry in the term-list, we can just analyze the term Biochemistry and identify Chemistry as its possible hypernym.

- There might be multiple terms for which no common part is an independent term but significant overlap exists among those, even more than once. In such cases we have introduced that overlapped part as our new term in the term-list. For example, we have Chocolate Pudding and Vanilla Pudding as two terms in our list but no entry for Pudding. Since we get overlapping in previous two terms with Pudding, we can consider Pudding as the possible hypernym of Chocolate Pudding and Vanilla Pudding. However, the problem is that we were getting some noise in the input due to the stopwords present in the list. For example, University of PlaceA and

University of PlaceB will have University and of as the subordinate-word hypernyms. Of cannot be a hypernym to some term. So we remove those subordinate-words which have only stopwords in them. Again, we had to deal with different morphological forms of the same word as hypernyms, for example science and sciences. For such instances, we checked if any one form is the part of our term list. If yes, we keep that form and remove others or we keep the lemmatized form otherwise.

C. Analysis of Result

Just as construction of suitable ontology from text, evaluation of an extracted ontology is not a simple task either. For this particular task, structural evaluation was done which includes the presence of cycles, the number of intermediate nodes compared to leaf nodes, and the number of over generic relations with the root node. The output relations were also evaluated against collected gold standards collected from WordNet and other well known, openly available taxonomies using evaluation measures like standard precision, recall and f-score.

Language	Precision	Recall	f-score
English	0.15	0.30	0.20
Dutch	0.16	0.22	0.19
French	0.17	0.25	0.20
Italian	0.13	0.20	0.19

Table 4.1 Average Precision, Recall, F-Score for Gold Standard Evaluation Across All Domains.

Table1 shows the average result of our system with respect to the gold standard evaluation for each language taking an average over all the domains. We had our focus on generating a hypernym tree for each term by providing the hypernyms of a term as next input to the system. This resulted in better recall but the precision of our system showed a visible decline compared to the baseline system for all the languages.

Subtask	Measures	Baseline	JUNLP
Monolingual (EN)	Cyclicity	**0**	3
	Structure (F&M)	0.0046	0.1498
	Categorisation (i.i.)	77.67	**377**
	Connectivity (c.c.)	36.83	53.17
	Gold standard comparison (Fscore)	**0.33**	0.20
	Domains	**6**	**6**
	Manual Evaluation (Precision)	n.a.	0.09
Multilingual (NL,FR,IT)	Cyclicity	**0**	**0**
	Structure (F&M)	0.0087	0.0155
	Categorisation (i.i.)	64.28	**178.22**
	Connectivity (c.c.)	40.5	34.89
	Gold standard comparison (Fscore)	**0.3133**	0.1921
	Manual Evaluation (Precision)	n.a.	0.2983

Table 4.2 Structural Evaluation for English and Other Languages

Table 2 shows the structural evaluation of the output produced by our system for different domains in English and other languages. The structural measures used for the evaluation are as follows:

- V: number of distinct vertices;

- E: number of distinct edges;

- c.c.: number of connected components;

- i.i.: intermediate nodes = V - L where L is the set of leaves

- cycles: YES = the taxonomy contains cycles, NO = the taxonomy is a Directed Acyclic Graph (DAG)

- Cumulative Fowlkes and Mallows Measure(FM): cumulative measure of the similarity of two taxonomies.

As we can see, though we have cycles present in relations of English language, all other language output is a DAG. We achieved better score in categorization due to high number of distinct vertices, edges and intermediate nodes obtained by our system, as mentioned in their detailed evaluation description[10].

We can try to improve our system's performance by making use of information available with Wikipedia dump other than the article texts such as infobox properties, redirect links, article titles, categories or other meta-information available. Also, provided a training set, we believe a bag-of-word model constructed within a specific context window can yield better overall results.

4.4 Context-based Relation Extraction Challenge

The system that we described in the previous section cannot be directly applied to the present situation as we do not have domain specific terms here, but named entities. And proper nouns do not generally hold a hypernym-hyponym relation with other proper nouns. Hence, it is not possible to get direct hypernym relations among the entities that we have extracted. Moreover, the linguistic pattern based approach which has been unanimously adopted to find out instances of these relations from text is not going to work in our task either. The Hearst patterns or the similar other patterns that other researchers have added over time are very much corpus specific and adhere to a particular writing style. On the contrary, we need to build a system that will work effectively on texts from various genres, which follow different literary style. These patterns won't be able to reach satisfactory recall score on datasets like Twitter, Blog or Review where the authors follow informal approach while adding content; as these texts do not strictly follow standard grammar rules and includes various foreign words and additional symbols. As direct taxonomic relations like hypernym-hyponym or meronym-holonym is not possible among named entities, we try to assign a score for each of these relations and for entity pair based on the context they frequently appear in. The relations we have considered to extract a score for are:

i. ***Hypernym-Hyponym***: Hyponymy shows the relationship between the more general terms (hypernyms) and the more specific instances of it (hyponyms). A hyponym is a word or phrase whose semantic field is more specific than its hypernym. The semantic field

[10] http://alt.qcri.org/semeval2016/task13/index.php?id=evaluation

of a hypernym, also known as a superordinate, is broader than that of a hyponym. An approach to the relationship between hyponyms and hypernyms is to view a hypernym as consisting of hyponyms. For instance, oak is a hyponym of tree, and animal is a hyponym of dog.

ii. ***Meronym-Holonym***: Meronym is a word that denotes a constituent part, member or substance of something that is complete in itself. On the contrary, Holonym is a word that denotes a thing that is complete in itself and whose part, member or substance is represented by another word. There are three types of meronyms:

Part meronym: a 'tire' is part of a 'car'

Member meronym: a 'car' is a member of a 'traffic jam'

Substance (stuff) meronym: a 'wheel' is made from 'rubber'

iii. ***Synonym-Antonym***: An antonym is a word that is the opposite meaning of another. It comes from the Greek words "anti" for opposite and "onym" for name. Since language is complex, people may at times, disagree on what words are truly opposite in meaning to other words. A synonym is a word that means the same, or almost the same, as another word.

We have used relations from WordNet in our algorithm. In the next sections, a basic idea about WordNet and the detailed implementation and sample result of this scheme is described along with the algorithm.

4.5 What is WordNet

WordNet is an on-line lexical reference system whose design is inspired by current psycholinguistic theories of human lexical memory. English nouns, verbs, and adjectives are organized into synonym sets, each representing one underlying lexical concept. Different relations link the synonym sets. WordNet's structure makes it a useful tool for computational linguistics and natural language processing. WordNet superficially resembles a thesaurus, in that it groups words together based on their meanings. However, there are some important distinctions. First, WordNet interlinks not just word forms—strings of letters—but specific senses of words. As a result, words that are found in close proximity to one another in the

network are semantically disambiguated. Second, WordNet labels the semantic relations among words, whereas the grouping of words in a thesaurus does not follow any explicit pattern other than meaning similarity.

In WordNet, a form is represented by a string of ASCII characters, and a sense is represented by the set of (one or more) synonyms that have that sense. WordNet contains more than 118,000 different word forms and more than 90,000 different word senses, or more than 166,000 (f,s) pairs. Approximately 17% of the words in WordNet are polysemous; approximately 40% have one or more synonyms. WordNet respects the syntactic categories noun, verb, adjective, and adverb—the so-called open-class words. For example, word forms like "back,'' "right,'' or "well'' are interpreted as nouns in some linguistic contexts, as verbs in other contexts, and as adjectives or adverbs in other contexts; each is entered separately into WordNet. It is assumed that the closed-class categories of English—some 300 prepositions, pronouns, and determiners—play an important role in any parsing system; they are given no semantic explication in WordNet. Inflectional morphology for each syntactic category is accommodated by the interface to the WordNet database. For example, if information is requested for "went," the system will return what it knows about the verb "go." On the other hand, derivational and compound morphology are entered into the database without explicit recognition of morphological relations. For example, "interpret," "interpreter,'' "misinterpret," "interpretation," "reinterpretation," "interpretive," "interpretative," and "interpretive dancing" are all distinct words in WordNet. A much larger variety of semantic relations can be defined between words and between word senses than are incorporated into WordNet. The semantic relations in WordNet were chosen because they apply broadly throughout English and because they are familiar— a user need not have advanced training in linguistics to understand them. WordNet includes the following semantic relations:

- Synonymy is WordNet's basic relation, because WordNet uses sets of synonyms (synsets) to represent word senses. Synonymy is a symmetric relation between word forms.

- Antonymy (opposing-name) is also a symmetric semantic relation between word forms, especially important in organizing the meanings of adjectives and adverbs.

• Hyponymy (sub-name) and its inverse, hypernymy (super-name), are transitive relations between synsets. Because there is usually only one hypernym, this semantic relation organizes the meanings of nouns into a hierarchical structure.

• Meronymy (part-name) and its inverse, holonymy (whole-name), are complex semantic relations. WordNet distinguishes component parts, substantive parts, and member parts.

• Troponymy (manner-name) is for verbs what hyponymy is for nouns, although the resulting hierarchies are much shallower.

• Entailment relations between verbs are also coded in WordNet.

Each of these semantic relations is represented by pointers between word forms or between synsets. More than 116,000 pointers represent semantic relations between WordNet words and word senses.

4.6 CRM: Contextual Relation Extraction Module

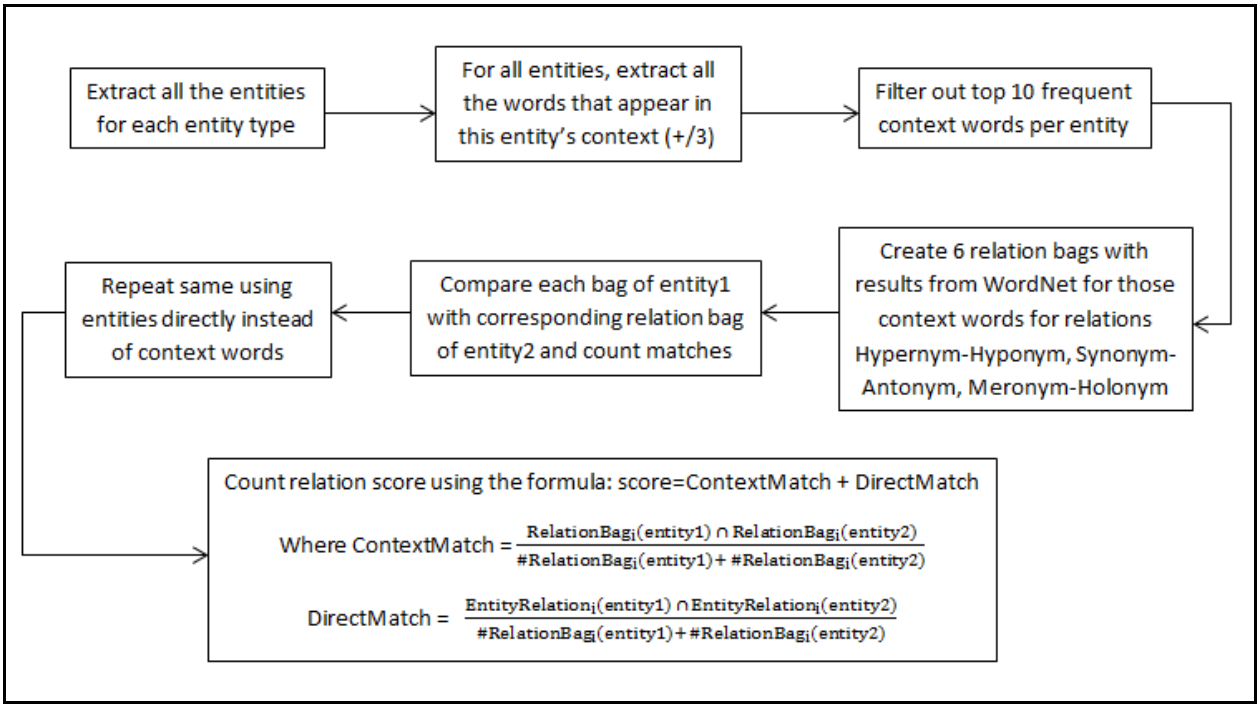

Fig 4.2 System Architecture for Calculating Taxonomic Relation Scores

To build a system that can be applied over all the domains we have considered, we need to keep the algorithm independent of the linguistic pattern or the contents of any particular dataset type. The steps of the algorithm are as follows:

A. For each dataset, create an entity bag of each type: PERSON, LOCATION, ORGANIZATION (and THING, EVENT for twitter).

B. For each entity, repeat step C to K

C. Crawl the entire dataset and extract all the words that appear in this entity's context (+/-3 word window) with their frequency.

D. Filter out top frequent 10 context words per entity. Do this for all entities.

E. Take another entity from same or different entity type that co-occur at least once with that entity and compare them based on six relations from WordNet: hypernym, hyponym, meronym, holonym, synonym and antonym.

F. Take all the 10 context words of an entity and add all the words obtained from WordNet for a particular relation (from the 6 mentioned above) to a bag, e.g. Put all the words that we can extract from WordNet for synonym relation of those context words to 'Synonym' bag.

G. Construct 6 different bags of words for the above-mentioned relation types for each of these two entities.

H. Compare each relation bag of one entity to corresponding relation bag of the other entity and count number of overlaps.

I. Calculate relation score for this particular entity pair using the formula:

$$\text{CW-Relation-Score}_i = \frac{\text{RelationBag}_i(\text{entity1}) \cap \text{RelationBag}_i(\text{entity2})}{\#\text{RelationBag}_i(\text{entity1}) + \#\text{RelationBag}_i(\text{entity2})}$$

CW: Context Word;

$i \in$ {hypernym, hyponym, synonym, antonym, holonym, meronym}

J. Also, along with the context word relation score calculations, do a similar calculation for entities themselves, e.g. compare among the hypernyms of entity1 and entity2 from WordNet and count the matches. Then divide the count with total number of hypernyms found for entity1 and entity2 combined.

$$\text{DM-Relation-Score}_i = \frac{\text{EntityRelation}_i(\text{entity1}) \cap \text{EntityRelation}_i(\text{entity2})}{\#\text{RelationBag}_i(\text{entity1}) + \#\text{RelationBag}_i(\text{entity2})}$$

DM: Direct Match;

i∈ {hypernym, hyponym, synonym, antonym, holonym, meronym}

K. In the last stage, we add these two scores CW-Relation-Score$_i$ and DM-Relation-Score$_i$ to obtain the final score of an entity-type for a relation.

L. Continue for other entity-type combinations as well. End.

For example, consider a situation where we have two entities IBM <ORG> and Bangalore <LOC> which co-occur in a single sentence. So we need to collect all the words that appear in these two entities' context and filter top-10 context words based on term-frequency count. If the context word bags for IBM is, BAG1: {office, main, research, employee, information, technology, company, development, cognitive, innovation} and for Bangalore, BAG2: {city, India, job, develop, prospect, beautiful, employee, information, technology, institute}; we will search in WordNet for hypernym, hyponym, meronym, holonym, synonym, antonym relation words of all the words of these two bags and create 6 different bags containing words for 6 different relations for each of these two entities. We count the number of overlapping words for each of these bags of IBM with Bangalore and normalize the count with total number of words present for that relation. In this way, we get a context word score for each of the relations. In a similar manner, instead of searching in WordNet for entries in BAG1 and BAG2, we directly search for the entities IBM and Bangalore. Though it is highly unlikely to have named entities in WordNet, we want to know if there is any such match present for direct match. Following the similar approach, we get 6 different direct match scores for each of those relations. Adding the context word scores and direct match scores, we obtain 6 final scores, one for each of the relations among the entities IBM and Bangalore.

4.7 Results & Observations

For each entity pair that co-occurs, we will have a score assigned for each of the six relations. The score will vary from 0 to 1, while 0 means the entities are not related at all and 1 means they are completely related for that particular relation. Here we present a sample snapshot of these relations for a few **blog** entities.

Entity-Pair	Hypernym	Hyponym	Antonym	Synonym	Meronym	Holonym
Berlin<LOC> - Karina <LOC>	0.0813	0.0295	0	0.2439	0.1463	0.1071
Germany<LOC>- Starbucks<ORG>	0.0683	0.0091	0	0.2626	0.2727	0
Sam<PER> - New York<LOC>	0.4860	0.2082	0.2222	0.4961	0.5	0
NFL<ORG> - League<ORG>	0.1738	0.5026	0	0.2306	0	0

Table 4.3 Snapshot of Taxonomic Relation Score for Blog Entities

Similarly scores for all other possible entity-pairs and for all the seven datasets were obtained. If we try to analyze what the results that are shown here infer:

1. *Karina* is a railway station in San Jose and *Berlin* is a city. So it can be easily assumed that they will probably have a meronym-holonym relation as a city may have a railway station. However, city and railway station are a top-level abstraction of the entities that we have here. Hence, we do not get a very high score in these fields, the score is not 0 either.

For the last entity-pair mentioned, we get two entities *NFL* and *League*. As we know that NFL is a type of League, we observe a high score for the relations Hypernym-Hyponym, whereas no score for Meronym-Holonym.

CHAPTER 5

NON-TAXONOMIC RELATION EXTRACTION

5.1 Introduction

Non-taxonomic relations are those for which a sensible and reusable taxonomy cannot be created using the relations, i.e. entities cannot be represented in a hierarchical fashion based on the relations among them. Non-taxonomic relation extraction is argued to be as one of the most difficult task and often neglected problem in ontology learning mechanism. It is this kind of relations which reveal more about a particular domain as taxonomic relations are restricted to some specific relations only, causing a hindrance to explore the domain in an exhaustive manner. As these relations vary immensely owing to the diverse nature of domains to extract an ontology from, it is very difficult to figure out how many type of relations are there to be extracted. The problem of non-taxonomic relation extraction can be categorized into two sub-problems:

(a) Non-taxonomic Relation Discovery: Identification of the domain concept pairs (C1, C2) such that some non-taxonomic relations hold from C1 → C2 or/and C2 → C1.
(b) Non-taxonomic Relation Labeling: Identification of labels for the non-taxonomic relations from C1 → C2 or/and C2 → C1.

As the domain changes, the type of relations changes rapidly; for example, a domain of Restaurants' data can have relations like:
<Restaurant_Name>- [situated in] → <Location_Name>
Or, <Restaurant_Name> - [specializes in] → <Cuisine_Name> etc.
But in the domain of Institutions, a few examples of non-taxonomic relations would essentially be as follows: <Institute_Name>- [offers] → <Course_Name>
Or, <Student> - [takes admission] → <Course_Name> etc.

This type of relations is atomic in nature, in the sense that they cannot be further structured into a definite form, like tree. Hence it is crucial that one chooses the type of relations to be extracted carefully to balance the coverage and correctness of the ontology created. Also, after discovering

a particular type of relation, the next thing to be kept in mind is to make sure it is appropriately labeled, as to carry an unambiguous meaning to the native users for further application and extension of that domain ontology.

5.2 Related Work

As already mentioned, the field of non-taxonomic relation extraction can be divided into two major subtasks which are Relation Discovery and Relation Labeling. From the journal of M. K. Wong et al (2014), we get to know that the research work in non-taxonomic relation discovery was first initiated by Maedche and Staab (2000) depending on a generalized association rule algorithm. In Text-to-Onto (Maedche and Staab,2000) and Text2Onto (Cimiano and Völker, 2005), the same association rule mining algorithm with a confidence measure is used to find out correlated concept pairs based on the linguistically related word-pairs which were discovered using shallow text processing. In Text2Onto, another algorithm helped to determine the level of abstraction most suited to describe those conceptual relationships by omitting the less appropriate or effective ones. The system claims to be evaluated and applied against the tourism and the insurance domain. The limitations of the system observed are that the performance greatly depends on the frequency of concepts in the dataset and it also returns pair of concepts even when no suitable relation was found.

The ASIUM ontology learning tool (Nedellec, 2000) approaches by the method of syntactic analysis to extract syntactic frames of verbs from given text documents. The system takes only the head nouns of phrases and links them with verbs while learning the syntactic roles. Adjectives and empty nouns are not considered in the process. The learning method in ASIUM is solely based on the observation of syntactic regularities in the context of words. Conceptual clustering is performed based on head nouns occurring with the same verb/verb phrases. However, for relation and cluster labeling, human interference could not be removed.

Another ontology learning tool, Hasti (Shamsfard and Barforoush, 2004) takes advantage of both morpho-syntactic and semantic analysis on unstructured input texts to extract lexical and ontological knowledge. The morpho-syntactic analysis predicts the features of unknown words that are encountered in the process and creates sentence structures. As opposed to co-occurrence

frequency analysis in non-taxonomic relation extraction in Text2Onto, Hasti is based on semantic analysis. Some of these templates are based on Hearst's patterns aimed at extracting hyponymy relations, while others are aimed at extracting other semantic relations. As the proposed approach was tested for Persian texts, the system is not applicable to English text directly. However, the approach can be followed to construct similar rules for other languages, changing those semantic templates or linguistic patterns.

OntoLearn (Velardi et al., 2005)makes use of a reduced set of FrameNet relations to train an available machine learning algorithm, TiMBL, which essentially is an open-source software package implementing several memory-based learning algorithms to extract relations between two concepts of a particular domain. The relations used are as follows: Material, Purpose, Use, Topic, Product, Constituent Parts and Attribute. The authors represented training instances as pair of concepts annotated with the appropriate conceptual relation, for example: [(computer #1, maker #2), Product]. Each concept is in turn represented by a feature vector where attributes are the concept's hypernyms in WordNet However, it is only capable of extracting a limited range of non-taxonomic relations. An evaluation of OntoLearn was conducted in the tourism and economy domain.

RelExt tool (Schutz and Buitelaar, 2005) employs a combination of both linguistic and statistical processing to find relations between the domain concepts and verbs. For linguistic processing, RelExt implemented a system to specify the dependency structure along with grammatical function mentioned, phrase structure, part-of-speech and lemmatization. This rich linguistic information is then explored to come up with a list of lemmatized head nouns and a list of lemmatized predicates. Numerous chi-square tests were carried out on the extracted lists to obtain the co-occurrence scores in determining the triplets that represent the non-taxonomic relations. As RelExt is directed toward ontology extension, the system relies on an already existing ontology for some domain, in order to map the head nouns that are highly relevant to corresponding concept labels.

When it comes to non-taxonomic relation labeling, most of the existing applications use verbs/verb phrases frequently occurring in the context of each concept pair association as label of the unnamed relations as verbs play a major role in communicating the sense of a valid sentence.

In the early version of Text-to-Onto, discovered non-taxonomic relations are labeled manually by human. Kavalec, Maedche and Svaték (2004) have built an extension to Text-to-Onto to include the automation of relation labeling. The basic idea of this research is to identify verbs that express relations between concepts by applying a heuristic statistical measure called above expectation based on conditional probability.

In Text2Onto (Cimiano and Völker, 2005), sub-categorization frames enriched by ontological knowledge and statistical information are extracted implementing shallow parsing approach to identify labels for the non-taxonomic relations. The confidence value for the relations extracted is carefully calculated based on the number of instantiations of that particular frame found in the data corpus at hand. Only those sub-categorization frames which are above a certain confidence value threshold are considered as labels for the non-taxonomic relations.
Sánchez and Moreno (Sánchez and Moreno, 2008) presented an approach using those verbs from sentences containing domain concept identifiers and then using search engine queries for relation labeling. The basic idea of this work is to extract a list of verbs related to each domain's concept and then calculate the relatedness of each concept to the verb. Subsequently, the concept-verb pairs are ranked and used as base for learning non-taxonomic related concepts from the web. The main advantage of this approach is that it actually thought of domain adaptation problem and came up with a independent one.

Weichselbraun et al. (2010) extract and aggregate verb vector from semantic relations identified in the corpus. Then, external structured data such as DBpedia and OpenCyc are used to refine the non-taxonomic labels. The proposed method would be needing 3 types of input: (a) domain ontology; (b) XML/RDF domain ontology containing unlabeled relations from the ontology learning framework and (c) a reusable "relation description" meta ontology which contains set of relation label to be used.

On the other hand, OntoLearn (Velardi et al., 2005) tool does not focus on verb as label. Instead, a trained classifier is applied to determine the labels of unlabeled relations from a reduced set of FrameNet relations (Material, Purpose, Use, Topic, Product, Constituent Parts and Attribute).

The labeling of non-taxonomic relations is very restricted to a few generic labels decided beforehand and also this approach is not particularly suited for more technical domains.
However, none of these systems has encountered with datasets that are extremely irregular in structure and topics covered. So we cannot apply these methods directly in our system which needs to deal with the structure variety present in the entire dataset.

In addition to the verb-based relation set, we also try to check if there exists any sentiment relation between two entities based on the polarity score among them, calculated over context word polarity scores from SentiWordNet. A lot of researchers have taken interest in this sentiment annotation task in recent days, adopting various methods, in both supervised and unsupervised research approaches. The main reason is that sentiment analysis has huge implementation scope in real life applications, though it is highly challenging and contains unusual sub-problems. In general, sentiment analysis has been investigated mainly at three levels:

Document level: The task at this level is to classify whether a whole opinion document expresses a positive or negative sentiment (Pang et al., 2002) (Turney et al., 2002).

Sentence level:The task at this level goes to the sentences and determines whether each sentence expressed a positive, negative, or neutral opinion. Neutral usually means no opinion. This level of analysis is closely related to subjectivity classification, which distinguishes sentences (called objective sentences) that express factual information from sentences (called subjective sentences) that express subjective views and opinions (Wiebe et al., 1999).

Entity and Aspect level: Both the document level and the sentence level analyses do not discover what exactly people liked and did not like. Aspect level performs finer-grained analysis. Aspect level was earlier called feature level (feature-based opinion mining and summarization) (Hu & Liu, 2004). Instead of looking at language constructs (documents, paragraphs, sentences, clauses or phrases), aspect level directly looks at the opinion itself. It is based on the idea that an opinion consists of a sentiment (positive or negative) and a target (of opinion). An opinion without its target being identified is of

limited use. Realizing the importance of opinion targets also helps us understand the sentiment analysis problem better.

5.3 Verb-Based Relation Extraction

The main challenge in our thesis, as already discussed, is that we are not dealing with a particular domain in focus. We are more concerned with different forms of unstructured texts, which in turn, can be based on multiple random topics. For example, one of our dataset is twitter data, which can be tweets on any random topic on earth, following to particular syntactic structure. In such cases, it is not possible for a particular domain expert to come up with a few fixed relations which will be able to cover all kind of relations that can possibly be in the domain. To address this issue at hand, we fully depend on the verbs occurring between two concepts. The advantage of using verbs in relation mining is that it works for both of the subtasks, relation discovery and relation labeling. There can broadly be two kinds of approaches to this method:

(a) Bottom-up approach - Here domain-related concepts appearing in the text collection are extracted first. Second, verbs occurring along with these concepts are being extracted and used as label for the relation.

(b) Top-down approach – At first verbs frequently appearing in the text collection are extracted and used as label for the relation. Then concept pairs occurring along with these verbs are being extracted as non-taxonomic concept pairs.

Previously it was considered that meaning is an essential outcome of the syntactic structure a text follows, and all other factors were ignored. Now, the researchers have come up with different methods to extract meaning from text which can broadly be categorized in two classes:

In Statistical approach, the distributional properties of words are studied through co-occurrence distributions of words. Semantic distances between the words are computed in order to determine the correlated concept pairs, which are considered to be potential candidates for non-taxonomic relations.

Lexico-syntactic approach is mainly based on string matching patterns based on text tokens and syntactic structure to discover non-taxonomic relations between a pair of co-occurring concepts in unstructured texts.

In our proposed method, we mainly adopted the Bottom-up syntactic approach where we use the already extracted Named Entities from previous step as concepts and try to extract relations among them based on co-occurrence using the freely available verb lexicon VerbNet.

5.3.1 Overview of VerbNet

VerbNet is a verb lexicon with syntactic and semantic information for English verbs, adhering to Levin verb classes (Levin, 1993) for systematic construction of lexical entries. It consists of approximately 5800 English verbs, and groups them into 274 first-level verb classes according to shared syntactic behaviors, thereby exploring the generalizations of verb behavior. Although the basis of VerbNet classification is syntactic, the verbs of a given class share semantic regularities as well because, according to Levin's hypothesis, the syntactic behavior of a verb is largely influenced by its meaning. This domain-independent lexicon takes full advantage of the systematic link between syntax and semantics that motivates these classes, and thus provides a clear and regular association between syntactic and semantic properties of verbs and verb classes (Kipper et al., 2000; Dang et al., 2000). To make this association explicit, a set of thematic roles is assigned to each syntactic argument in a given verb class as well as some selectional restrictions to each of these theme roles.

5.3.2 Components of a Verb Class in VerbNet

Class Hierarchy – Contains the verbs in hierarchical fashion, i.e. the main sense of a verb as root and the derived senses as children or subclasses. Each individual subclass, in turn, may include one or more subclasses that broadly generalize to that particular sense.

Members – Contains the list of actual English verbs belonging to a specific class or subclass. Most of them are mapped to entries in other lexical resources including FrameNet (Baker et al., 1998), WordNet (Miller, 1990; Fellbaum, 1998), and Xtag (XTAG Research Group, 2001), among which VerbNet works as a connecting tool deriving meaning from all.

Roles – Thematic roles is basically the semantic relationship between a predicate and its arguments, categorized into 23 distinct roles. VerbNet makes use of a hierarchical thematic

roleset, in which, for each class, the roles that are thought to be core to the verb members' behavior are listed. Brief description of the roles needed for our task is given below.

Actor:	Used for some communication classes (e.g., Chitchat-37.6, Marry-36.2, Meet-36.2) when both arguments can be considered symmetrical (pseudo-agents).
Agent:	Generally a human or an animate subject. Used mostly as a volitional agent, but also used in VerbNet for internally controlled subjects such as forces and machines.
Beneficiary:	The entity that benefits from some action. Used by such classes asBuild-26.1, Get-13.5.1, Performance-26.7, Preparing-26.3, and Steal-10.5. Generally introduced by the preposition `for', or double object variant in the benefactive alternation.
Location, Destination, Source:	Used for spatial locations.
Destination:	End point of the motion, or direction towards which the motion is directed. Used with a `to' prepositional phrase by classes of change of location, such as Banish-10.2, and Verbs of Sending and Carrying. Also used as location direct objects in classes where the concept of destination is implicit (and location could not be Source), such as Butter-9.9, or Image impression-25.1.
Source:	Start point of the motion. Usually introduced by a source prepositional phrase (mostly headed by `from' or `out of'). It is also used as a direct object in such classes as Clear-10.3, Leave-51.2, and Wipe

	instr-10.4.2.
Location:	Underspecified destination, source, or place, in general introduced by a locative or path prepositional phrase.
Experiencer:	Used for a participant that is aware or experiencing something. In VerbNet it is used by classes involving Psychological Verbs, Verbs of Perception, Touch, and Verbs Involving the Body.
Instrument:	Used for objects (or forces) that come in contact with an object and cause some change in them. Generally introduced by a `with' prepositional phrase. Also used as a subject in the Instrument Subject Alternation and as a direct object in the Poke-19 class for the Through/With Alternation and in the Hit-18.1 class for the With/Against Alternation.
Material and Product:	Used in the Build and Grow classes to capture the key semantic components of the arguments. Used by classes from Verbs of Creation and Transformation that allow for the Material/Product Alternation.
Material:	Start point of transformation.
Product:	End result of transformation.
	Used for participants that are undergoing a process or that have been affected in some way. Verbs that explicitly (or implicitly) express changes of state have Patient as their usual direct object. We also

Patient:	use Patient1 and Patient2 for some classes of Verbs of Combining and Attaching and Verbs of Separating and Disassembling, where there are two roles that undergo some change with no clear distinction between them.
Recipient:	Target of the transfer. Used by some classes of Verbs of Change of Possession, Verbs of Communication, and Verbs Involving the Body. The selection restrictions on this role always allow for animate and sometimes for organization recipients.
Time:	Class-specific role, used in Begin-55.1 class to express time.
Topic:	Topic of communication verbs to handle theme/topic of the conversation or transfer of message. In some cases, like the verbs in the Say-37.7 class, it would seem better to have `Message' instead of `Topic', but we decided not to proliferate the number of roles.

Table 5.1: List of Thematic Roles and Example Classes From VerbNet

Selectional Restrictions - Each of these 23 thematic roles listed in a class can be further characterized by a few selectional restrictions, which provide more information about the nature of a given role. The hierarchy of these selectional restrictions is as follows:

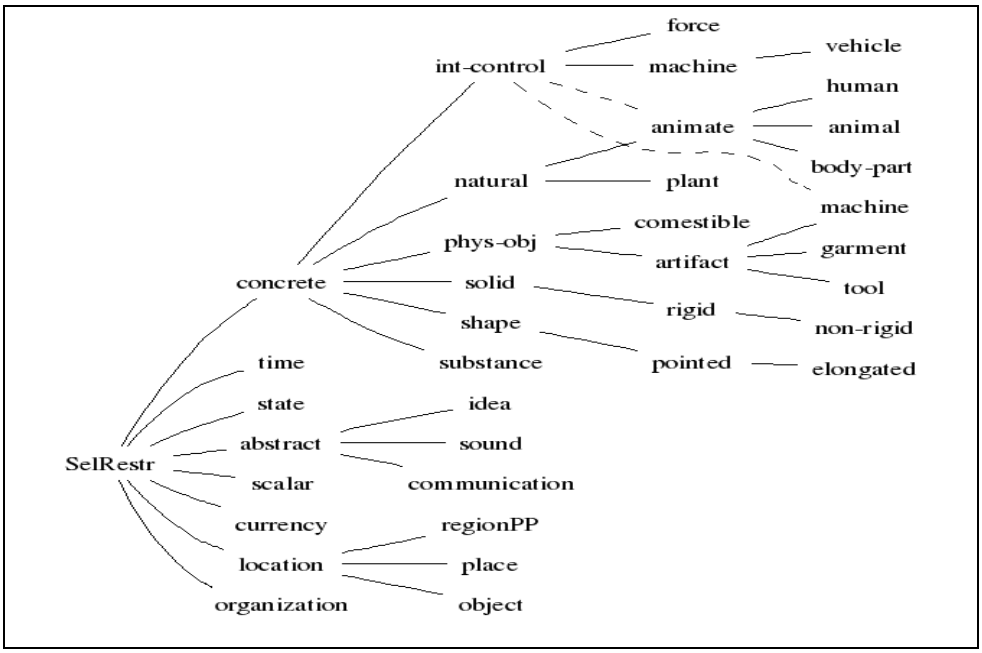

Fig 5.1 Hierarchy of Selectional Restrictions in VerbNet

Frames – Provides a description of the different syntactic behavioral characteristics of verb classes and alternations of syntactic frame patterns allowed for the members of the class. The Frames section consists of syntactic constructions, example sentences, and the semantic roles mapped to syntactic arguments. Semantic predicates are also taken into consideration in this section, to give an idea about how the participants are involved in the event.

All the Verb Classes are numbered according to shared semantics and syntax, and classes sharing a top-level number (9-109) have corresponding semantic relationships. For instance, verb classes related to putting, such as put-9.1, put_spatial-9.2, funnel- 9.3, etc. are all assigned to the class number 9 and related to moving an entity to a location. Classes that share a top class can also be divided into subclasses. Class numbers 1-57 are drawn directly from Levin's (1993) classification. Class numbers 58-109 were developed later in the work of Korhonen & Briscoe (2004). To be noted, the verb types of the later classes are less general, as most of these classes have a one-to-one relationship between verb type and its corresponding verb class. The top class of the hierarchy consists of syntactic constructions and semantic role labels that are shared by all verbs in this class. VerbNet subclasses inherit features from the top class but specify further syntactic and semantic commonalities amongst their verb members. These can include additional

syntactic constructions, further selectional restrictions on semantic role labels, or new semantic role labels. Because subclasses inherit content from their parent classes, they can be considered as members of the parent class but with more specific features. If a subclass is directly dominated by a parent class and the same parent class also directly dominates another subclass, then those two subclasses are sisters to one another. Sister classes do not share features.

5.3.3 Proposed Approach

As we have this major disadvantage of not knowing the domain theme beforehand, and texts from a single genre can be on any arbitrary topic and even multiple topics as well, there is no way we can design a flexible set of possible relations that needs to be extracted, even after rigorous discussion of a domain expert and an ontology developer. Hence we make use of the verbs present at hand; as it has been seen that verb is an essential part of making sense out of a randomly drawn unstructured text. But it won't be enough to just identify a lot of relations from the text. We needed to find a way to cluster them, in due course which will be providing us with more insight about those entities. The logic being, there is a high chance of concepts or entities being similar in some dimension, which may or may not be very definitive one, if they belong to the same cluster. VerbNet, being such an efficient and free-to-use resource, which was not used for this kind of task before, was chosen as the primary tool to explore the dataset for verb based non-taxonomic relation clustering.

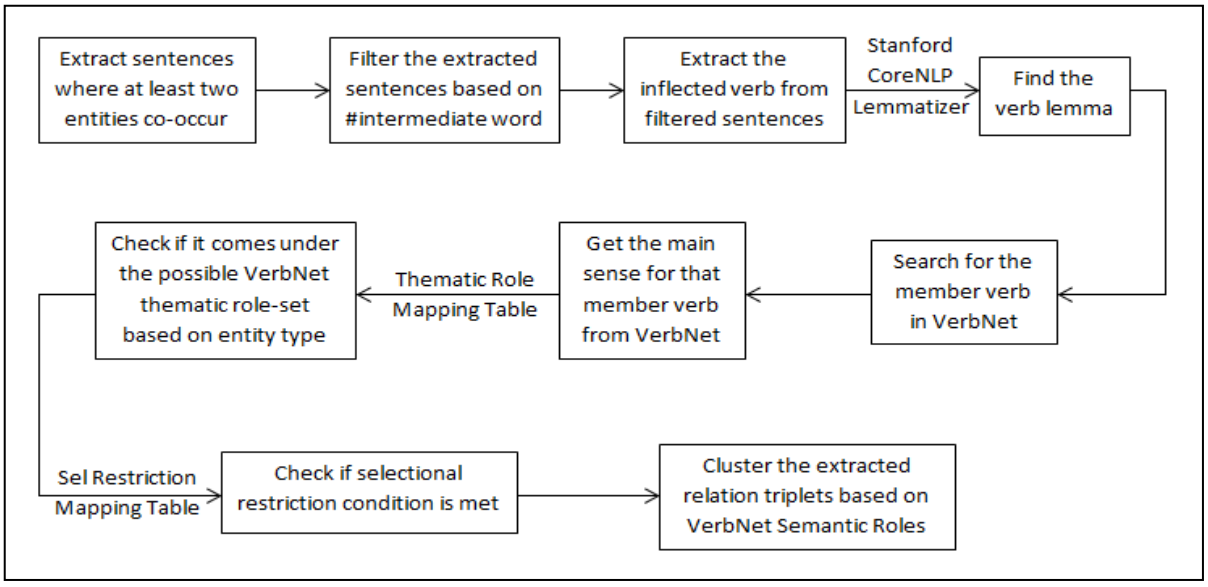

Figure 5.2 System Architecture for VerbNet Based Relation Extraction Module

The actual method goes through a series of preprocessing and evaluation tests, which are briefly described below:

- The sentences in which two entities of same or different types occur together are extracted from each domain.
    - *For example,*
        1. *Being the Chairperson of our company, Mr. Rajnish Patel <Person> donated the most for the cause of Teach for India <Organization>.*
        2. *Aladin<Person> got married in the palace hall, in the presence of thousands of eminent people from all over the world, who are friends with Sultan<Person>.*

- Only the sentences in which the entities occur in a window size of maximum 10 are filtered out, i.e. there will be at most 10 words in between the entities. This is done to restrict the number of irrelevant entity-relationships that are extracted.
    - *For the previous example, only the first one (Being the Chairperson of our company, Mr. Rajnish Patel <Person> donated the highest sum for the cause of Teach for India <Organization>) is filtered out as number of words in between the entities is 8, which is less than 10. Whereas, this intermediate window size for second sentence is 23, which is much greater than 10.*
    - *We can see that a relation like Aladin-<marry>-Sultan would have carried wrong information. Hence in order to achieve better precision, we should ignore the co-occurrence of entities which are apart by a large intermediate-word window. The size of this window was determined through empirical study.*

- We tag the sentences with Stanford CoreNLP POS tagger and only the verbs are extracted from those filtered sentences.
    - *From the example above, the verb extracted is 'donated'.*

- Using Stanford CoreNLP lemmatizer, extracted verbs are transformed into the corresponding lemma, so that they can be used to search VerbNet to acquire a greater

understanding. Lemmatization usually uses a vocabulary and does morphological analysis of words, normally aiming to remove inflectional endings only and to return the base or dictionary form of a word, which is known as the lemma. If confronted with the token *saw*, stemming might return just *s*, whereas lemmatization would attempt to return either *see* or *saw* depending on whether the use of the token was as a verb or a noun. The two may also differ in that stemming most commonly collapses derivationally related words, whereas lemmatization commonly only collapses the different inflectional forms of a lemma.

- *So the verb extracted in raw form 'donated' is now in its lemmatized version: 'donate'.*

- Next, the lemmatized form of the verb is searched for in VerbNet and the main sense, of which this verb-form is a member verb is extracted as the relation sense.
    - *'donate' is a member verb of the sense 'contribute'(contribute-13.2-1-1), which actually belongs to the main sense 'contribute-13.2'*

- After we get the main sense of that particular verb which appeared in our dataset, we need to check if it belongs to the set of thematic roles which are allowed for those particular types of entities this verb is appearing in the context of. The thematic role mapping table is mentioned below:

Entity Type	Possible Thematic Roles from VerbNet
Person	Actor, Agent, Co-Agent, Beneficiary, Experiencer, Patient, Co-Patient, Recipient
Location	Location, Source, Destination, Initial_Location
Organization	Agent, Co-Agent, Beneficiary, Recipient
Thing	Instrument, Material, Product
Event	Time, Topic

Table 5.2 Thematic Role Mapping

- *The sentence in the example had entities of type 'Person' (Mr. Rajnish Patel) and 'Organization' (Teach for India).*

  So the set of possible thematic roles for the verb encountered is:

  *{*Actor, Agent, Co-Agent, Beneficiary, Experiencer, Patient, Co-Patient, Recipient*}*

  Thematic Roles associated with the verb sense 'contribute-13.2' are: AGENT, THEME and RECIPIENT. Among them, the roles AGENT and RECIPIENT in present in the set of possible thematic roles for the entity types at hand.

  So, we consider the sense 'contribute' for relation after this phase.

- We try to further filter the verbs using selectional restrictions associated with each thematic role. We did a mapping for entity type and possible selectional restriction in the same manner as of the thematic roles. The mapping is provided below:

Entity Type	Possible Selectional Restriction Chains from VerbNet
Person	Concrete->natural->animate->human
Location	Location->regionPP
Organization	Organization
Thing	Concrete->phys-obj->artifact->tool Concrete->solid->rigid Concrete->substance Concrete->int-control->machine
Event	Time

Table 5.3 Selectional Restriction Mapping

- *The sentence in the example had entities of type 'Person' (Mr. Rajnish Patel) and 'Organization' (Teach for India). So the possible selectional restrictions would be:*

  {Human, Organization}

  The verb-sense 'contribute-13.2' has thematic roles AGENT, THEME and RECIPIENT. The selectional restrictions applicable for these theme-roles are:

--Agent [+animate | +organization]

--Recipient [+animate | +organization]

As these selectional restrictions 'animate' and 'organization' are included in the set of restrictions allowed in our approach, we will finally consider the verb sense 'contribute' to be a possible relation sense between two entities 'Mr. Rajnish Patel' (Person) and 'Teach for India' (Organization)

- In a similar manner, all possible verb-based relations among all possible entity pairs of any two types are extracted. As the latest version of VerbNet has 274 first level classes in which all the other verbs are categorized into based on the sense they are communicating, we would have got 274 different clusters for '*entity-relation-entity'* triplets. But it's practically impossible to visualize that many classes for extracted relations. Hence we incorporated the concept of thematic role in our clustering algorithm. As there are at most 23 thematic roles, any relation extracted from any of the domain will come under one or more of those 23 classes only. Compared to the previous 274 classes, it is much easier for users to understand the sense of those relations and use or extend them effectively for future use.

5.3.4 Results and Observations

We note down the number of relations and corresponding number of clusters for each entity-pair type in each of the seven domains.

Domain Name	VerbNet Relation Statistics		
	Entity Pair Type	**#Relations**	**#Clusters**
	Person-Person	16	6
	Organization-Organization	11	8
	Person-Location	34	8

Twitter	Person-Organization	7	6
	Person-Thing	208	18
	Person-Event	3	3
	Organization- Thing	21	6
	Organization-Event	4	2
	Location-Thing Location-Location Location-Event Location-Organization Thing-Thing Event- Event Thing-Event	0	0
Wikipedia Article	**Entity Pair Type**	**#Relations**	**#Clusters**
	Person-Person	50	13
	Person-Location	97	18
	Person-Organization	119	12
	Location-Location Organization-Organization Location-Organization	0	0
	Entity Pair Type	**#Relations**	**#Clusters**
	Person-Person	198	3

Blog	Person-Organization	72	3
	Location-Location Organization-Organization Person-Location Location-Organization	0	0
NEWS	**Entity Pair Type**	**#Relations**	**#Clusters**
	Location-Location	0	0
	Organization-Organization	51	11
	Person-Location	125	19
	Location-Organization	34	6
Review	**Entity Pair Type**	**#Relations**	**#Clusters**
	Organization-Organization	23	7
	Person-Location	1605	17
	Person-Organization	154	12
	Location-Organization	2	2
	Location-Location	0	0
	Entity Pair Type	**#Relations**	**#Clusters**
	Person-Person	703	18

Contemporary Literature	Person-Location	876	19
	Person-Organization	56	2
	Location-Organization Organization-Organization Location-Location	0	0
Mythology	**Entity Pair Type**	**#Relations**	**#Clusters**
	Person-Person	106569	23
	Organization-Organization	19	6
	Person-Location	22387	23
	Person-Organization	7305	22
	Location-Organization	65	12
	Location-Location	0	0

Table 5.4 Verb based Relation Statistics for All Domains

As one can observe, the number of relations extracted per entity-pair varies with dataset and also with entity-pair types. A PERSON-type entity-rich 'mythology' dataset extracts a large number of relations for all the entity-pairs that include PERSON. In the 'contemporary literature' domain, it is seen that we get relations only for the pairs which includes at least one PERSON type entities. Similarly, for other domains as well, it can be witnessed that our system failed to extract any relation for a few entity types. For example, the count of entities for THING-EVENT or EVENT-EVENT entity pairs in twitter, ORGANIZATION-ORGANIZATION in blog, LOCATION-LOCATION in review is nil. This can be explained with the notion that our system only detects relation when two different entities of an entity-pair co-occur in a single sentence. Moreover, one of these entities must be the subject of the verb, and also satisfy the thematic role

and selectional restriction mapping in VerbNet done for its corresponding entity type. So we can conclude from our observations, that the entity-pairs having nil relation counts in all the domains either never appears together in a single sentence, or the sentences they co-occur in, do not contain a verb which is appropriate for the entity types they belong to, in terms of thematic role assigned in VerbNet or selectional restrictions for that role.

If we try to note the variations in terms of relation-counts, we will notice a striking difference among the datasets like twitter and contemporary literature or mythology. While the relation counts for Twitter data is very less or nil, we find the same count to be very high for mythology dataset. One possible reason for this outcome is that we deal with short texts in twitter, and hence the probability of two entities co-occurring together is low. A similar trend can be witnessed in Blog dataset as well. But in case of literature and especially mythological texts, our system acquires more instances of these, even in the review dataset as well; because they deal with characters and so chances of entities appearing together is higher.

We have seen some noisy outcomes as well. To list a few:

i. Harry Potter – <celebrate> – Harry_P

   One can easily notice the problem here: both the entities actually refer to same character Harry Potter, but as the forms are different, system will treat them as different entities and try to find relations among them. This kind of issues can be removed if we implement an additional entity linking module before relation extraction phase and merge all forms of same entity together to indicate a single actual reference.

ii. Aladin – <marry> – Sultan

   Going by the famous short story, it can be said that this particular relation is not correct. However, according to our system output, these two entities are connected via this verb because they appear in each other's context along with the verb. This particular problem can be avoided by adding an additional check to keep only the relations if two participating entities are related with *subject-verb-object* frame. But this will significantly reduce the recall for other type of entity-pairs other than the ones which contains at least one PERSON type entity, as

entities like LOCATION or ORGANIZATION hardly appears as a subject of verb in a sentence.

iii. Consider the sentence: *"Jason was not ready to agree with Julia this time."* – For this particular sentence, the relation *Jason – Julia – <agree>* will be extracted which is directly opposite to the sense here. For these cases, a negation checker needs to be included in the system.

Following are a few sample relations from different datasets:

Domain	Entity-Pair	Relation	Cluster
Twitter	Organization – Event Person – Event Person – Thing	Police – Riot – <reach> @GKollings_4 – Christmas – <love> Nick Andersen – British – <talk>	Agent, Destination Actor Agent, Co-Agent
Wikipedia Articles	Person – Location Person – Person	Albert – New York – <grow> Hector – Achilles – <confront>	Agent, Loacation Agent, Theme
Blog	Person – Person	Chris – Veitch – <amuse>	Stimulus, Result
Review	Person – Location Person – Organization	Chan – Africa – <marry> king – court – <urge>	Agent Agent, Recipient
NEWS	Organization – Organization	Communist – Laos – <use>	Agent, Value
Contemporary Literature	Person – Location Person – Person	Aladdin – China – <settle> Aladdin – Prince – <poison>	Agent, Goal Instrument

Mythology	Person – Location Person – Person	Madras – army – <reach> Shikhandi – Kritavarma – <resign>	Agent, Destination Agent, Co-Agent, Recipient, Source

Table 5.5 Sample Verb Relations from Different Domains

A snapshot containing a few of such verb relations among a subset of entities from contemporary literature domain is presented below. Here the entity set is: {Aladdin, Mustafa, Fatima, Sultan}. The thematic-role based clusters that we see here are: Theme, Result, Stimulas and Co-Agent. And the relations they contain are 'amuse', 'dub', 'marry', 'accompany' etc.

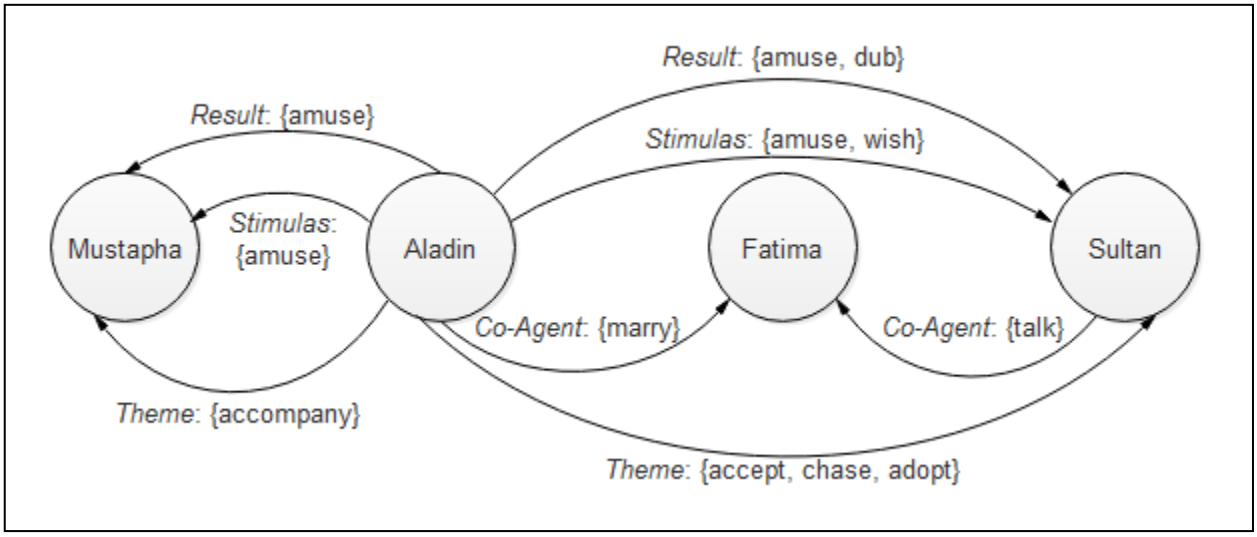

Fig. 5.3 Sample Verb Relations and Thematic Role Based Cluster

5.4 Sentiment Based Relations

Opinions and its related concepts such as sentiments, evaluations, attitudes and emotions are the subjects of study of sentiment analysis and opinion mining, which is one of the most trending topics of recent researches in the field of Natural Language Processing. Sentiment classification is a part of opinion mining activity involved in determining the overall sentiment, opinions, sentiments, evaluations, appraisals, attitudes and emotions towards entities such as products, services, organizations, individuals, issues, events, topics, and their attributes. It represents a

large problem space. There are also many names and slightly different tasks, e.g., sentiment analysis, opinion mining, opinion extraction, sentiment mining, subjectivity analysis, affect analysis, emotion analysis, review mining, etc. However, they are now all under the umbrella of sentiment analysis or opinion mining. The classes of an opinion mining activity can be either of these two types:

Two distinct classes: *Positive* and *Negative*. Additionally, there can be another class representing the neutral ones, i.e. the ones with no prominent sentiment. For example: opinion on delivery system of a particular company in e-commerce business.

Ranks according to a spectrum of possible opinions, for example 0 to 5 stars for films or restaurants in review data.

> *Liu defines a sentiment or opinion as a quintuple- "<o_j, f_{jk}, so_{ijkl}, h_i, t_l>, where o_j is a target object, f_{jk} is a feature of the object o_j, so_{ijkl} is the sentiment value of the opinion of the opinion holder h_i on feature f_{jk} of object o_j at time t_l, so_{ijkl} is +ve,-ve, or neutral, or a more granular rating, h_i is an opinion holder, t_l is the time when the opinion is expressed."[19]*

Most of the researches in the field of opinion mining have explored datasets containing different articles from web, product, services or film reviews, microposts such as twitter or blogging data etc. Main advantage of such web-based datasets is that there is a continuous flow of new data getting added to the existing ones giving the researchers ample scope to use it in their advantage. Tasks like Sentiment span detection, Cross Domain and Close Domain Sentiment Analysis, Sentiment-wise Document Classification etc. have already been carried out with different datasets in different domains. However, during relation discovery phase of ontology engineering, sentiment-based relations have been ignored till date in spite of having the potential to exhibit meaningful relations among entities. The types of entities we are dealing with are: Person, Location, Organization, Thing and Event. Interestingly, we have observed that some of these types can display meaningful sentiment based relations among them. To give a brief idea of our approach, we can say that analyzing all the sentences with appropriate tools or sentiment analysis module of our own where two entities co-occur, we can comment whether those two entities are related to each other in a positive or negative sense, if at all there is any relation. For example, in

the Microposts twitter dataset, we have tags Person and Thing, which includes products, groups, sports etc. If a specific tweet is about a 'Person' and a 'Thing' entity type, there is a possibility that we can comment how that 'Person' feels about that particular 'Thing', whether in positive, neutral or in a negative manner, analyzing the context words they co-occur with.

> Sample: *In a recent interview, @Sharapovasaya Tennis is the love of her life.*
>
> In this particular tweet, we have one <Person> type entity, i.e. '@Sharapova' and one <Thing> type entity, 'Tennis'. By using appropriate modules to analyze the context words of these two entities, we can draw a conclusion about the relation between '@Sharapova' and 'Tennis'.

We have used SentiWordNet 3.0.0 to build a module that will calculate the effective sentiment score of the context words associated with any entity-pair. A brief overview of the SentiWordNet 3.0.0 is given below.

5.4.1 Overview of SentiWordNet

The SentiWordNet is a freely available lexical resource which contains a list of English terms which have been credited a score of positivity and negativity. SentiWordNet provides this information which is extracted and utilized in different algorithms to produce an overall score of the text snippets or target words at hand and thus predicting the expression expressed in the text or document. Each synset***s*** is credited three numerical scores ***Pos(s)***, ***Neg(s)***, and ***Obj(s)*** in the range [0.0 to 1.0] which tell us how "positive", "negative" or "objective" (i.e., neutral) the terms enclosed in the synset are. Different senses of the same term may thus exhibit different opinion-related characteristics. For example, in SentiWordNet 1.0 the synset*[estimable(J,3)]*, conforming to the sense *"may be computed or estimated"* of the adjective *estimable*, has an Obj score of 1.0 (and Pos and Neg scores of 0.0), while the synset*[estimable(J,1)]* corresponding to the sense *"deserving of respect or high regard"* has a Pos score of 0.75, a Neg score of 0.0, and an Obj score of 0.25. To keep it brief, SentiWordNet extends the WordNet by addition of subjectivity information (+ or -) to every word in the database. Since same words can have different meanings with respect to the part of speech being represented, SentiWordNet was designed by ranking subjectivity of all terms/synsets according to the part of speech the term carries in that sense. The parts of speech represented by the SentiWordNet are adjective, noun, adverb and verb

which are represented respectively as 'a', 'n', 'r', 'v'. The lexicon has five columns, the part of speech, the offset – which is a numerical ID, that when matched with a particular part of speech, identifies a synset from WordNet; positive score, negative score (bottom from 0 to 1) and synset terms that includes all terms belonging to a particular synset.

Four different versions of SentiWordNet have been discussed in publications: *SentiwordNet 1.0* (Esuli&Sebastiani, 2006), *SentiwordNet 1.1* (Esuli&Sebastiani, 2007b), *SentiwordNet 2.0* (Esuli, 2008) and *SentiwordNet 3.0* (Esuliet al., 2010). Since versions 1.1 and 2.0 have not been discussed in widely known formal publications, we focus on the other two and main differences of SentiWordNet 1.0 and 3.0 are the following:

5.4.2 Proposed Approach

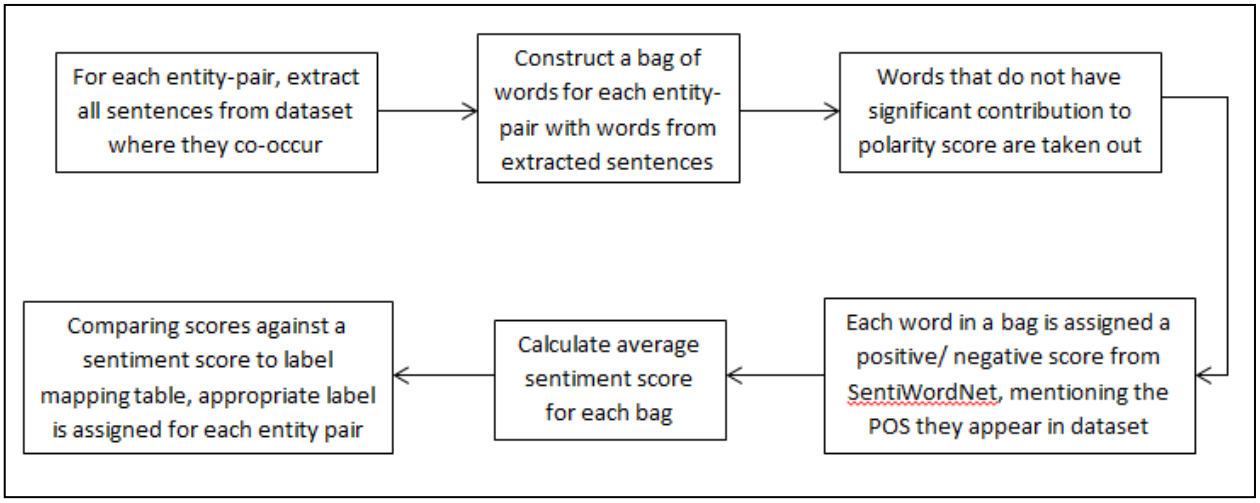

Fig. 5.4 System Diagram for Sentiment Based Relation Extraction Module

The primary disadvantage that we have faced while trying to examine this kind of relations among entities is that we had no annotated data ready to extract features from and train a model. Hence, the module we employed solely depends on SentiWordNet scores. We have diverse datasets containing text from web such as tweets, blog data, IMDb review data, as well as wiki article corpus, contemporary literature corpus and mythological text. We needed to build a system which will be simple enough that it can be applied to all formats of texts without user interference or domain-wise modification reducing any kind of bias as much as possible. Following is the stepwise approach we followed to compute the sentiment score between two entities:

- At first, we take an entity-pair of same or different type and extract all the sentences from dataset where they co-occur.
    - *For example,*
        1. *Raman<Person> was angry at Sophia<Person>, as she didn't respond to his mail in time.*
        2. *Another victory, Wimbledon<Location> loves you back @rogerfederar<Person>!*

        These are two sample sentences that can be taken out from the dataset as they contain entity-pair of same (Person-Person) or different (Person-Location) type.

- Excluding the entities themselves and any other entity that might appear in the context, all other words that appear in those sentences are collected in their lemmatized form (using Stanford CoreNLPlemmatizer) to build a bag of words for that particular entity pair.
    - *For the above two examples,*
        1. *Entity pair: Raman-Sophia*
           Bag of Words: is, angry, at, as, she, do, not, respond, to, his, mail, in, time
        2. *Entity pair: Wimbledon-@rogerfederar*
           Bag of Words: another, victory, love, you, back

- We do a SO (Subject-Object) analysis on that bag of words that were collected parsing the entire dataset We only keep the words that can have significant contribution (with score >0.25 or <-0.25) on polarity score.
    - Applying this on these two entity-pairs, we get:
        1. *Entity pair: Raman-Sophia*
           Filtered Words: angry, not, respond, time
        2. *Entity pair: Wimbledon-@rogerfederar*
           Filtered words: victory, love

- We are left with only a handful of words that have negative/positive polarity scores associated with them. We extract scores from SentiWordNet for each of the words separately mentioning the POS they appear with in that sentence.
    - *Extracting the scores for the filtered words for both the sentences:*
        1. *Entity pair: Raman-Sophia*

           Word Scores:

           Angry= -0.875

           Not= -0.625

           Respond= 0.625

           Time=0.5
        2. *Entity pair: Wimbledon-@rogerfederar*

           Word Scores:

           Victory= 0.375

           Love= 0.625
- We take the average over all the context word sentiment scores for that particular entity-pair. Comparing the score with prefixed mapping, we assign a sentiment relation between those entities.

Average Sentiment Score	Sentiment Label
score>=0.75	Very Positive
0.25<score<0.75	Positive
-0.25<score<0.25	Neutral
-0.75<score<-0.25	Negative
score<=-0.75	Very Negative

Table 5.5 Mapping Table - Sentiment Score v/s Sentiment Label

- *Extracting the scores for the filtered words for both the sentences:*
    1. *Entity pair:* ***Raman-Sophia***

       *Sentiment Score Average:****-0.09***

       Sentiment Label: ***Neutral***
    2. *Entity pair:* ***Wimbledon-@rogerfederar***

Sentiment Score Average: ***0.5***

Sentiment Label: ***Positive***

- Finally, after labeling each entity pair that co-occurs, we put them into three distinct classes: Positive, Negative and Neutral

5.4.3 Results and Observation

The number of entities was not same for all the datasets that we have considered for present work. Hence, if we carry out a comparative study among domains based on the number of relations acquired for each dataset, it would be highly biased. To remove this dependency, we have taken top 50 entities from each entity type (i.e. PERSON, LOCATION, ORGANIZATION for all and in addition, THING and EVENT for Twitter dataset only) for each dataset i.e. Twitter, Blog, Review, Wikipedia Article, Contemporary Literature and Mythology. The number of sentiment based relations will vary for each dataset because of the irregular distribution of entities over different datasets, and also, will be hugely dependent on the style of writing that a particular genre follows. A table containing the statistics for sentiment based relations over all datasets is listed below, which will help us coming up with a constructive analysis of the performance of our system and the results obtained.

Dataset	Relation Statistics			
	Entity-Pair Type	#co-occurrence	#Positive	#Negative
	Person-Person	268	6	5
	Person-Location	103	10	12
	Person-Organization	68	7	5
	Person-Thing	49	3	3
	Person-Event	37	5	2
	Location-Location	253	17	23
	Location-Organization	327	24	28

Twitter	Location-Thing	192	18	16
	Location-Event	327	28	6
	Organization-Organization	146	13	12
	Organization-Thing	94	8	5
	Organization-Event	96	18	10
	Thing-Thing	151	18	7
	Thing-Event	71	8	0
	Event-Event	249	17	18
Wikipedia Article	Entity-Pair Type	#co-occurrence	#Positive	#Negative
	Person-Person	8112	18	10
	Person-Location	5243	31	22
	Person-Organization	1008	37	18
	Location-Location	7840	69	31
	Location-Organization	9408	82	43
	Organization-Organization	9984	44	16
Blog	Entity-Pair Type	#co-occurrence	#Positive	#Negative
	Person-Person	50	12	9
	Person-Location	1450	70	205
	Person-Organization	48	10	2

	Location-Location	75	8	5
	Location-Organization	50	11	6
	Organization-Organization	44	3	10
Review	Entity-Pair Type	#co-occurrence	#Positive	#Negative
	Person-Person	45950	328	275
	Person-Location	8400	198	130
	Person-Organization	408	138	81
	Location-Location	3950	53	45
	Location-Organization	648	58	58
	Organization-Organization	224	14	13
NEWS	Entity-Pair Type	#co-occurrence	#Positive	#Negative
	Person-Person	6350	17	17
	Person-Location	900	47	44
	Person-Organization	300	51	66
	Location-Location	3550	46	32
	Location-Organization	1500	101	50
	Organization-Organization	2770	57	32
	Entity-Pair Type	#co-occurrence	#Positive	#Negative

Contemporary Literature	Person-Person	5550	44	14
	Person-Location	650	14	8
	Person-Organization	679	13	10
	Location-Location	250	3	3
	Location-Organization	85	3	1
	Organization-Organization	189	0	0
Mythology	Entity-Pair Type	#co-occurrence	#Positive	#Negative
	Person-Person	24451	241	133
	Person-Location	448	79	40
	Person-Organization	130	45	12
	Location-Location	574	6	0
	Location-Organization	80	9	1
	Organization-Organization	28	3	0

Table 5.6 Sentiment based Relation Statistics for All Domains

- ***Twitter*:** Our model fetched decent number of relations for a few entity-pair set and remarkably low for the others. While trying to get to the root of this problem, we realized that there are not many entities which co-occur in a single sentence of a tweet as the word limit is restricted in Twitter. Moreover, frequent presence of foreign words and special symbols make it difficult to predict a sentiment score using SentiWordNet as it only includes words from WordNet, i.e. without any spelling error or contraction. In future, this particular problem can be addressed by introducing a spell-correction module before relation extraction phase. Another reason for this low turnover of relations can be the absence of a detailed discussion on a specific topic as tweets belong to the category of

microposts. We have a large number of entities at hand, but each of them is associated with only a small fraction of total posts. Therefore, we do not get sufficient number of words which can have significant contribution to polarity score.

Relation Type	Entity Pairs	Context Words
POSITIVE	Chelsea <ORG>-World Cup<EVE>	Sign, do, well, under, 20, World, Cup
NEGATIVE	Murdoch<PER>-hackgate<EVE>	Corrupt, crumble, empire
NEUTRAL	British<THING>-cnnbrk<THING>	White, woman, think

Table 5.7 Sample Sentiment Relations for Twitter Domain

- ***Blog*:** Similar to twitter, blogs are also dynamic user-generated data which depends greatly on the author profile, i.e. age, gender, region of stay, language proficiency etc. However, compared to the former, spelling contractions or usage of special symbols is much less in blogs as it follows a more conventional manner of writing. It managed to produce more number of relations than twitter, but not as high as compared to review or mythology. The primary reason being the span of the discussion for a specific entity. Compared to twitter, blogs can dedicate more words against any particular entity as there are no restrictions on word count as microposts, but being short articles, blogs do not have the scope of carrying a topic for too long as review or a Wikipedia article.

Relation Type	Entity Pairs	Context Words
POSITIVE	John<PER> - Switzerland<LOC>	Better, part, lounge
NEGATIVE	Phil<PER>-Sheldon<PER>	Want, awkward, stop, smile, look, straight
NEUTRAL	Germany<LOC>-	Cup, Berlin, city

	World<LOC>	

Table 5.8 Sample Sentiment Relations for Blog Domain

- ***Review*:** The review dataset we have collected consists of original movie reviews on IMDb. Though reviews have no word count restrictions, it follows a more or less informal approach than blog writing. It also includes foreign words or special symbols or abbreviations like twitter, though in a lesser frequency. One huge advantage that review dataset has, along with being an entity-rich dataset, is that we get a larger count of context words to analyze for sentiment relation score, which carry significant polarity scores themselves as they are user feedbacks for products and hence includes personal opinion. As an obvious outcome to these two advantages, we get most number of sentiment-based relations from this domain.

Relation Type	Entity Pairs	Context Words
POSITIVE	John<PER>-America<LOC>	Goofy, Bald, Villainous, Role
NEGATIVE	William<PER>-Edward<PER>	norton , russ , paul, tough, say, something, nice, ethnocentric
NEUTRAL	FBI<ORG>-washington<LOC>	Assistant, special, agent-in-charge, anthony

Table 5.9 Sample Sentiment Relations for Review Domain

- ***NEWS*:** The news dataset that we are considering here is from the press reportage category of Brown corpus. Newspaper reporting, being a passively written content, follows a conventional literary style and do not generally reflects author's emotion while generating that content. As these articles go through a tedious proofreading process, one can hardly find any spelling or grammatical errors in these texts. News articles generally deal with a decent number of named entities and spare an entire article to talk about a particular issue. As an obvious result, number of sentences where two or more entities co-

occur is high and our system finds enough context words to calculate the polarity score between entities.

Relation Type	Entity Pairs	Context Words
POSITIVE	Dallas<LOC> - Texas<LOC>	Big, City, Like, Effort, Representative, large, money, fill, provide, better, get
NEGATIVE	Moscow<LOC> - White House<ORG>	House, Reaction, Bitter, Exchange
NEUTRAL	Jack<PER> - Chicago<LOC>	Stadium, see, county

Table 5.10 Sample Sentiment Relations for NEWS Domain

- ***Wikipedia Article*:** Unlike all other user generated documents in web that we are handling, Wikipedia articles are written in extremely professional manner, following a formal writing style and ideally without any spell-errors or abrupt contractions. This unique quality of these texts makes them an unbiased dataset to work with. Unlike twitter, these articles are elaborated and so can spend more words describing a particular entity, thus giving rise to number of context words to analyze sentiment. These very factors help our system strike a balance in terms of number of polarized relations for this domain.

Relation Type	Entity Pairs	Context Words
POSITIVE	America<LOC>- Lincoln<PER>	debate , generally, consider, most, famous, political, carefully
NEGATIVE	Europe<LOC>- Robert<PER>	Draw, ethnography, philology, analyze, society, differentiate
NEUTRAL	Lincoln<PER>- Court<ORG>	Appear, front, Illinois, Supreme

Table 5.11 Sample Sentiment Relations for Wikipedia Article Domain

- ***Contemporary Literature*:** This collection contains texts from various authors. So the writing habit and usage of words varies throughout the domain texts, albeit maintaining a formal literary approach. As the distribution of entities is not regular for all entity types, one can observe vast difference among the number of entities extracted for different entity-pairs. For example, number of entities for ORGANIZATION-ORGANIZATION pair is nil. This is because there were not enough ORGANIZATION entries in this dataset and those few which were there were scattered throughout different pieces of text, thus hampering the co-occurrence figure.

Relation Type	Entity Pairs	Context Words
POSITIVE	Santa<PER>-San Francisco<LOC>	merry, nice, fine, wife, night, Christmas, dear , sure , happy
NEGATIVE	Chris<PER>-Mulligan<ORG>	Get, stomachache, march
NEUTRAL	Mulligan<ORG>-Alley<ORG>	Say, cool, icicle, Shantytown

Table 5.12 Sample Sentiment Relations for Contemporary Literature Domain

- ***Mythology***: The writing manner this particular dataset is focused on differs from all the other ones to a great degree. Usage of complex words and unusual sentence structure are the primary difficulties faced by our system to analyze this data. Moreover, there was a significant lack in number of LOCATION and ORGANIZATION entities, though there was ample number of PERSON entities and as the entire document is part of the same epic, we found a large number of context words for all the entities we were considering. That led the performance to obtain a large number of relations for PERSON-PERSON entities while the yield was visibly low for other entity-pairs which doesn't contain at least one PERSON-type entity.

Relation Type	Entity Pairs	Context Words
POSITIVE	Kshatriyas<PER>-Bhagadatta<PER>	High, soul, heroic
NEGATIVE	Duryodhana<PER>-Army<ORG>	Solicit, thee, offer, leadership, deprive, run, drag, arrow, battle, keen, shaft, slay, mighty, slaughter, killing, destroyer
NEUTRAL	Earth<LOC>-Sarasvati<LOC>	Sky , Cardinal, Subsidiary, point, compass , Trees, mother, god

Table 5.13 Sample Sentiment Relations for Mythology Domain

CHAPTER 6

CONCLUSION & FUTURE SCOPES

The main challenge of present thesis was twofold, as we needed to extract entity and relations from multiple datasets of different genres displaying diverse characteristics in terms of text structure, vocabulary used, influence of external agents like region, time, culture, platform etc. In addition to this, the datasets we are considering can be on any random topic or theme. An entity identification and classification system is built that recognizes and classifies top-level named entities from texts. Next, we try to find out the possible taxonomic and non-taxonomic relations that can exist among these entities.

6.1 Entity Extraction

One of the main aims for this module was to build a model that will be able to extract named entities of types PERSON, LOCATION, ORGANIZATION from texts of different genres and domains (e.g., twitter, blog, news, reviews, literature etc.). Working on two hypotheses, single training set hypothesis and mixed training set hypothesis, we extracted domain-dependent and domain-independent feature values from texts and sent them to a CRF based classifier and finally modifying the annotated tags using gazetteer lists. Though we achieved modest results in both the hypotheses, a few following measures can be taken for further improvements.

- Instances from mixed training set that have greater impact on the results with respect to a test set will be given greater weightage during its training so that the classifier may work better for that particular domain.
- Inclusion of entity type specific features like trigger words or n-gram prefix-suffix to increase the recall of our system.
- We will try to devise methodologies for distinguishing closely related entity pairs such as CHARACTERS-PERSONS and THING-PRODUCT along with the identification of various other entity types such as EVENTS etc.

6.2 Taxonomic Relation Extraction

One of our main aims was to build a system that will work similarly in all the domains; we kept our approach as simple as possible. As we have only considered named entities, it practically doesn't make sense to try to extract direct taxonomic relations among them. Hence, we built a module that assigns scores to each pair of entities for six different taxonomic relations (Synonym, Antonym, Hypernym, Hyponym, Holonym, Meronym) based on context word analysis. Other improvements that can be implemented in future are:

1. We can think of mining some entity specific relations such as Family or Genealogical Relations for PERSON-type entities, hierarchical relations for LOCATION such as an instance of a city can be within an instance of a country. However, recall for these types of entity-pairs will be very low as the texts in those documents are not focused on a single theme on entity.
2. In future, we can improve these scores by introducing additional features and/or weighted context word scores.

6.3 Non-Taxonomic Relation Extraction

We have proposed two different non-taxonomic relation extraction schemes: one is verb based, and another is entity-pair sentiment based. In the verb based relation extraction module, we extract relations based on entity co-occurrence and the verb present, further filtering and clustering the extractions based on thematic role and selectional restrictions from VerbNet. In sentiment based relation module, we try to predict a polarity label for each pair on entity that co-occurs, based on the context words of co-occurrence.

6.3.1 Verb-based Relations

The VerbNet-based domain independent non-taxonomic relation extraction system that we have built has successfully extracted an ample number of meaningful relations from each domain and for various entity pairs.

1. However, some noisy results were present in the relation set which needs one or two additional modules to get removed. Removing these types of noises will increase the precision of the system.

2. To improve the recall, we can pass the raw data through a co-reference resolution model so that the pronouns get replaced by the nouns they represent, and we get more number of hits while searching for sentences containing multiple named entities.

3. In addition to these, further filtering of clusters can contribute more to sense disambiguation of the relations extracted.

6.3.2 Sentiment-based Relations

We have implemented a model based on simple intuitions to extract sentiment relations from six different types of dataset, containing texts about different topics. Though we have received a satisfactory number of relations for each of the datasets, we have figured out a few techniques that might improve the performance in future.

1. We can increase the precision by introducing a co-reference resolution module for datasets that consist of long articles as usage of pronouns increases with the length of a passage. If we can successfully replace the pronouns with appropriate entities, we will be able to fetch more context words from those sentences which in turn, will increase the accuracy in calculating the sentiment polarity score.

2. For the current work, we have only considered entity pairs, i.e. two entities at a time. It is possible to analyze each sentence separately and extract all the entities it contains (possibly more than two) and predict a sentiment score by taking all of them into consideration.

3. Analyze part of the datasets manually to extract relations and then using it as gold standard training, build a classifier model using more features in addition to SentiWordNet scores to improve performance.

RESEARCH PUBLICATIONS

1. Ghosh, S., Maitra, P. and Das, D. Feature Based Approach to Named Entity Recognition and Linking for Tweets. In Proceedings, 6th Workshop on Making Sense of Microposts (#Microposts2016): Big things come in small packages, Montreal, Canada 11th of Apr, 2016.

2. Maitra, P. and Das, D. JUNLP at SemEval-2016 Task 13: A Language Independent Approach for Hypernym Identification. Proceedings of the 10th International Workshop on Semantic Evaluation. Association for Computational Linguistics. 2016.

3. Maitra, P., Ghosh, S. and Das, D. Authorship Verification – An Approach based on Random Forest. In proceedings of PAN at CLEF, 2015.

REFERENCES

- Alfonseca, E., Manandhar, S. 2002. An Unsupervised Method for General Named Entity Recognition and Automated Concept Discovery. In Proc. International Conference on General WordNet

- Aman, S., Szpakowicz, S. "Identifying Expressions of Emotion in Text". V. Matousek, P. Mautner (eds.), Proc 10th International Conf. on Text, Speech and Dialogue TSD 2007, Plzeň, Czech Republic, Lecture Notes in Computer Science 4629, Springer, 196-205, 2007.

- Aman, S., Szpakowicz, S. "Using Roget's Thesaurus for Fine-grained Emotion Recognition". Proc Third International Joint Conf. on Natural Language Processing IJCNLP 2008, Hyderabad, India, 296-302, 2008.

- Ando, R. and Zhang, T. 2005. A Framework for Learning Predictive Structures from Multiple Tasks and Unlabeled Data. In Journal of Machine Learning Research 6 (2005), pages 1817–1853.

- Arnold, A., Nallapati, R., and Cohen, W. W. 2008. Exploiting Feature Hierarchy for Transfer Learning in Named Entity Recognition. In Proceedings of 46th Annual Meeting of the Association of Computational Linguistics (ACL'08), pages 245-253.

- Asahara, M., Matsumoto, Y. 2003. Japanese Named Entity Extraction with Redundant Morphological Analysis. In Proc. Human Language Technology conference - North American chapter of the Association for Computational Linguistics.

- Assadi, H. Construction of a regional ontology from text and ist use within a documentary system. In N. Guarino (ed.), Formal Ontology in Information Systems, Proceedings of FOIS-98, Trento, Italy, 1999, pages 236–249, 1999.

- Baccianella, S., Esuli, A. and Sebastiani, F., SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining. Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10), European Language Resources Association (ELRA), Valletta, Malta, 2010.

- Banko, M. and Etzioni, O. 2008. The tradeoffs between open and traditional relation extraction. In ACL-HLT.

- Baroni, M. and Lenci, A. 2010. Distributional memory: A general framework for corpus-based semantics. Computational Linguistics, 36.

- Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. A theory of learning from different domains. Machine Learning, 79:151–175, 2010.

- Ben-David, S., Blitzer, J., Crammer, K., and Pereira, F. Analysis of representations for domain adaptation. In Advances in Neural Information Processing Systems (NIPS), 2007.

- Bikel, D. M., Miller, S., Schwartz, R., Weischedel, R. 1997. Nymble: a High-Performance Learning Name-finder. In Proc. Conference on Applied Natural Language Processing.

- Blitzer, J., McDonald, R., and Pereira, F. 2007. Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL'07), pages 440-447.

- Blitzer, J., McDonald, R., and Pereira, F. Domain adaptation with structural correspondence learning. In Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2006.

- Borthwick, A., Sterling, J., Agichtein, E., Grishman, R. 1998. NYU: Description of the MENE Named Entity System as used in MUC-7. In Proc. Seventh Message Understanding Conference.

- Bouaud J, Bachimont B, Charlet J, Zweigenbaum P. ICCS'94 Workshop on Knowledge Acquisition using Conceptual Graph Theory. College Park, MD: University of Maryland; 1994. Acquisition and structuring of an ontology within conceptual graphs; pp. 1–25.

- Brachman RJ. What Is-a Is and Isn't - an Analysis of Taxonomic Links in Semantic Networks. Computer. 16(10):30–36.

- Cimiano P, Völker J (2005) Text2Onto: a framework for ontology learning and data-driven change discovery. In: Proceedings of the 10th international conference on applications and natural language to databases(NLDB '05), pp 227–238

- Daum´e III, H. Frustratingly easy domain adaptation. In Proc. of the Annual Meeting of the Association of Computational Linguistics (ACL), 2007.

- Daum´e III, H. and Marcu, D. Domain adaptation for statistical classifiers. Journal of Artificial Intelligence Research, 26:101–126, 2006.

- Daum´e III, H., Kumar, A., and Saha, A. Coregularization based semi-supervised domain adaptation. In Advances in Neural Information Processing Systems (NIPS), 2010.

- Do, Q. X., Roth, D., Constraints based Taxonomic Relation Classification, In proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 1099–1109, MIT, Massachusetts, USA, 9-11 October 2010

- Esuli, A. and Sebastiani, F., SENTIWORDNET: A high-coverage lexical resource for opinion mining. Technical Report 2007-TR-02, Istituto di Scienza e Tecnologiedell'Informazione, ConsiglioNazionaledelleRicerche, Pisa, IT, 2007b.

- Esuli, A. and Sebastiani, F., SENTIWORDNET: A publicly available lexical resource for opinion mining. In Proceedings of the 5th Conference on Language Resources and Evaluation (LREC'06), pages 417–422, Genova, IT, 2006.

- Esuli, A., Automatic Generation of Lexical Resources for Opinion Mining: Models, Algorithms, and Applications. Ph.D. thesis, Scuola di Dottorato in Ingegneria"Leonardo da Vinci", University of Pisa, Pisa, IT, 2008.

- Fall A. School of Computing Science. Simon Fraser University; 1996. Reasoning with taxonomies.

- Faure, D. and Nedellec, C. A corpus-based conceptual clustering method for verb frames and ontology acquisition. In In LREC workshop on Adapting lexical and corpus resources to sublanguages and applications, Granada, Spain, Mai 1998., 1998.

- Fellbaum, C., 1998. WordNet: An Electronic Lexical Database. MIT Press.

- Fellbaum, C., WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press, 1998.

- Freitag. 2004. Trained Named Entity Recognition Using Distributional Clusters. In Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP 2004)

- Gimpel, K., Schneider, N., O'Connor, B., Das, D., Mills, D., Eisenstein, J., Heilman, M., Yogatama, D., Flanigan, J., and Smith, N. A. Part-of-speech tagging for twitter: Annotation, features, and experiments. In ACL2011, page 42, 2011.

- Gruber T. R. A translation approach to portable ontology specifications. Knowledge Acquisition, 1993, 5(2) : 199~ 220

- Guarino N. The role of identity conditions in ontology design. Spatial Information Theory. 1999. 1661:221–234.

- Guo, H., Zhu, H., Guo, Z., Zhang, X., Wu, X. and Su, Z. In Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the ACL, pages 281–289, Boulder, Colorado, June 2009.

- Hall, M., Frank, E. , Holmes, G., Pfahringer, B. , Reutemann, P. , and Witten, I. H. The weka data mining software: An update. SIGKDD Explorations, 11:231, 2009.

- Harabagiu, S. M., Miller, G. A., and Moldovan, D. I., WordNet 2: A morphologically and semantically enhanced resource. In Proceedings of the ACL Workshop on Standardizing Lexical Resources(SIGLEX'99), pages 1–8, College Park, US, 1999.

- Hearst, M.A.. Automatic acquisition of hyponyms from large text corpora. In Proceedings of the 14th International Conference on Computational Linguistics. Nantes, France, 1992.

- Hobbs, J. The generic information extraction system. In Proceedings of the Fifth Message Understanding Conference (MUC-5), Morgan Kaufmann, 1993., 1993.

- Hu, M. and Liu, B., Mining and summarizing customer reviews. In Proceedings of ACM SIGKDD International Conference on KnowledgeDiscovery and Data Mining (KDD-2004). 2004.

- Jiang, J. and Zhai, C. 2006. Exploiting Domain Structure for Named Entity Recognition. In Proceedings of HLT-NAACL 2006, pages 74–81.

- Jiang, J. and Zhai, C. Instance weighting for domain adaptation in nlp. In Proc. of the Annual Meeting of the Association of Computational Linguistics (ACL), 2007.

- Kietz, J.-U. , Volz, R., and Maedche, A. Semi-automatic ontology acquisition from a corporate intranet. In International Conference on Grammar Inference (ICGI-2000), to appear: Lecture Notes in Artificial Intelligence, LNAI, 2000.

- Kozareva, Z., Riloff, E., and Hovy, E. 2008. Semantic class learning from the web with hyponym pattern linkage graphs. In ACL-HLT

- Li, W. and McCallum, A. 2005. Semi-supervised sequence modeling with syntactic topic models. In Proceedings of Twenty AAAI Conference on Artificial Intelligence (AAAI-05).

- Liu, Bing, Sentiment Analysis and Opinion Mining, 5th Text Analytics Summit, Boston,June 1-2, 2009

- Maedche A, Staab S (2000) Discovering conceptual relations from text. In: Proceedings of the 13th european conference on, artificial intelligence (ECAI-2000), pp 321–325

- Maedche A, Staab S (2000) The text-to-onto ontology learning environment. In: Software demonstrationat the 8th international conference on conceptual structures (ICSS-2000), pp 14–18

- Maedche, A. and Staab, S. Mining ontologies from text. In Proceedings of EKAW-2000, Springer Lecture Notes in Artificial Intelligence (LNAI-1937), Juan-Les-Pins, France, 2000. Springer, 2000.

- Maedche, A., Pekar, V. and Staab, S., Ontology Learning Part One --- on Discovering Taxonomic Relations from the Web, Web Intelligence, Springer Berlin Heidelberg, pp 301—319, 2003.

- Martin Kavalec, Alexander Maedche, Vojtech Svatek (2004), Discovery of Lexical Entries for Non–Taxonomic Relations in Ontology Learning, In proceedings of 30th Conference on Current Trends in Theory and Practice of Computer Science, Czech Republic, pp 249-256, 2004

- McCallum, A., Li, W. 2003. Early Results for Named Entity Recognition with Conditional Random Fields, Features Induction and Web-Enhanced Lexicons. In Proc. Conference on Computational Natural Language Learning.

- Mikheev, A., Moens, M., Grover, C. 1999. Named Entity Recognition without Gazetteers. In Proc. Conference of European Chapter of the Association for Computational Linguistics

- Miller, G. A., WordNet: A Lexical Database for English. Communications of the ACM Vol. 38, No. 11: 39-41, 1995.

- Miller, G. A., WordNet: A Lexical. Database for English, COMMUNICATIONS OF THE ACM November 1995/Vol. 38, No. 11. 39, 1995

- Miller, S., Guinness, J., and Zamanian , A. 2004. Name Tagging with Word Clusters and Discriminative Training. In Proceedings of HLT-NAACL 04

- Morin, E. Automatic acquisition of semantic relations between terms from technical corpora. In Proc. of the Fifth International Congress on Terminology and Knowledge Engineering - TKE'99, 1999.

- Moro, A. , Cecconi, F. , and Navigli, R. Multilingual word sense disambiguation and entity linking for everybody. In 13th International Semantic Web Conference, Posters and Demonstrations (ISWC 2014), page 25, 2014.

- Moro, A., Raganato, A., and Navigli, R. Entity linking meets word sense disambiguation: a unified approach. Transactions of the Association for Computational Linguistics (TACL), page 231, 2014.

- Nadeau, D., Turney, P., Matwin, S. 2006. Unsupervised Named Entity Recognition: Generating Gazetteers and Resolving Ambiguity. In Proc. Canadian Conference on Artificial Intelligence.

- Neches, R., Fikes, R. E. Gruber T. R. Enabling technology for knowledge sharing. AI Magazine，1991，12(3)：36~56

- Nedellec C (2000) Corpus-based learning of semantic relations by the ILP system, Asium. In: CussensJ, Dzeroski S (eds) Proceedings of learning language in logic. Springer, Berlin, pp 259–278

- Nothman, J. and Ringland, N. and Radford, W. and Murphy, T. and Curran, J. R., Learning multilingual named entity recognition from Wikipedia, Artificial Intelligence, Elsevier, Volume- 194, pages- 151-175, 2012.

- Palmer, D. D., Day, D. S. 1997. A Statistical Profile of the Named Entity Task. In Proc. ACL Conference for Applied Natural Language Processing.

- Pang, B. and Lee, L., A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts, Proceedings of the ACL, 2004.

- Pang, B., Lee, L., and Vaithyanathan, S. Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of Conference on Empirical Methods in Natural Language Processing(EMNLP-2002). 2002

- Pantel, P. and Pennacchiott, M. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In ACL, pages 113–120

- Ponzetto, P. and Strube, M., 2007. Deriving a large scale taxonomy from wikipedia. AAAI.

- Proceedings, 6th Workshop on Making Sense of Microposts (#Microposts2016): Big things come in small packages, Montreal, Canada 11th of Apr, 2016.

- Rizzo, G. and Erp M. van., Making Sense of Microposts (#Microposts2016) Named Entity recognition and Linking (NEEL) Challenge. In 6th Workshop on Making Sense of Microposts (#Microposts2016), Montreal, 2016

- Rizzo, G. and van Erp, M. Making Sense of Microposts (#Microposts2016) Named Entity rEcognition and Linking (NEEL) Challenge. In 6th Workshop on Making Sense of Microposts (#Microposts2016) [1], 2016.

- Schutz A, Buitelaar P (2005) RelExt: a tool for relation extraction from text in ontology extension. In:Proceedings of the 4th international semantic web conference, pp 593–606

- Sekine, S. 1998. Nyu: Description of the Japanese NE System Used For Met-2. In Proc. Message Understanding Conference.

- Shamsfard M, Barforoush AA (2004) Learning ontologies from natural language texts. Int J Hum ComputStud 60(1):17–63

- Shinyama, Y., Sekine, S. 2004. Named Entity Discovery Using Comparable News Articles. In Proc. International Conference on Computational Linguistics

- Snow, R., Jurafsky, D., and Ng, A. Y. Learning syntactic patterns for automatic hypernym discovery. In NIPS. 2005.

- Snow, R., Jurafsky, D., and Ng, A. Y. Semantic taxonomy induction from heterogenous evidence. In ACL. 2006.

- Stamatatos, E., Daelemans, W., Verhoeven, B., Juola, P., López-López, A., Potthast, M. and Stein, B., Overview of the Author Identification Task at PAN 2015, In Proceedings of PAN, CLEF-2015, CEUR Workshop Proceedings, Toulouse, France, 2015

- Suchanek, F. M., Kasneci,G. , and Weikum, G. 2007. Yago: A Core of Semantic Knowledge. In WWW.

- Sun, B. "Named entity recognition Evaluation of Existing Systems" Norwegian University of Science and Technology Department of Computer and Information Science, Masters' Thesis. 2010

- Sánchez D, Moreno A (2008) Learning non-taxonomic relationships from web documents for domainontology construction. Data Knowl Eng 64(3):600–623

- Turney, Peter D. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL-2002).2002.

- Velardi P, Navigli R, Cucchiarelli A et al. (2005) Evaluation of OntoLearn, a methodology for automatic learning of domain ontologies. In: Buitelaar P, Cimiano P, Magnini B (eds) Ontology learning from text:methods, applications and evaluation. IOS Press, Amsterdam, pp 92–106

- Weichselbraun A, Wohlgenannt G, Scharl A (2010) Refining non-taxonomic relation labels with externalstructured data to support ontology learning. Data Knowl Eng Eng 69(8):763–778

- Whitelaw, C., Patrick, J. 2003. Evaluating Corpora for Named Entity Recognition Using Character-Level Features. In Proc. Australian Conference on Artificial Intelligence

- Wiebe, J., Bruce, R. F. and O'Hara, T. P., Development and use of a gold-standard data set for subjectivity classifications. In Proceedings of the Association for Computational Linguistics (ACL-1999).1999.

- Wong, M. K., Syed Sibte Raza Abidi, Jonsen, I.D., A multi-phase correlation search framework for miningnon-taxonomic relations from unstructured text. Knowledge and Information Systems(2014) 38:641–667

- Woods WA. Understanding subsumption and taxonomy: A framework for progress. In: Sowa JF, editor. Principles of Semantic Networks. San Mateo, CA: Morgan Kaufmann; 1991. pp. 45–94.

- Yee Seng Chan and Hwee Tou Ng. 2007. Domain Adaptation with Active Learning for Word Sense Disambiguation. In Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL'07).