2016

Knowledge-Driven Implicit Information Extraction

Pathirage Dinindu Perera
Wright State University

Knowledge-driven Implicit Information Extraction

A dissertation submitted in partial fulfilment
of the requirements for the degree of
Doctor of Philosophy

By

SUJAN PERERA
B.Sc, University of Colombo, 2008

2016
Wright State University

WRIGHT STATE UNIVERSITY
GRADUATE SCHOOL

AUG 16, 2016

I HEREBY RECOMMEND THAT THE DISSERTATION PREPARED UNDER MY SUPERVISION BY Sujan Perera ENTITLED Knowledge-driven implicit information extraction BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF Doctor of Philosophy.

Amit Sheth, Ph.D.
Dissertation Director

Michael Raymer, Ph.D.
Director, Computer Science and Engineering Ph.D. Program

Robert E.W. Fyffe, Ph.D.
Vice President for Research and Dean of the Graduate School

Committee on
Final Examination

Krishnaprasad Thirunarayan, Ph.D.

Michael Raymer, Ph.D.

Pablo N. Mendes, Ph.D.

ABSTRACT

Perera, Sujan. PhD., Department of Computer Science and Engineering, Wright State University, 2016. Knowledge-driven Implicit Information Extraction.

Implicit information in unstructured text can be efficiently extracted by bridging syntactic and semantic gaps in natural language usage and by augmenting information extraction techniques with relevant domain and contextual knowledge.

Natural language is a powerful tool developed by humans over hundreds of thousands of years. The extensive usage, flexibility of the language, creativity of the human beings, and social, cultural, and economic changes that have taken place in daily life have added new constructs, styles, and features to the language. One such feature of the language is its ability to express ideas, opinions, and facts in an implicit manner. This is a feature that is used extensively in day to day communications in situations such as: 1) expressing sarcasm, 2) when trying to recall forgotten things, 3) when required to convey descriptive information, 4) when emphasizing the features of an entity, and 5) when communicating a common understanding.

Consider the tweet *'New Sandra Bullock astronaut lost in space movie looks absolutely terrifying'* and the text snippet extracted from a clinical narrative *'He is suffering from nausea and severe headaches. Dolasteron was prescribed'*. The tweet has an implicit mention of the *entity Gravity* and the clinical text snippet has implicit mention of the *relationship* between medication *Dolasteron* and clinical condition *nausea*. Such implicit references of the entities and the relationships are common occurrences in daily communication and they add unique value to conversations. However, extracting implicit constructs has not received enough attention in the information extraction literature. This dissertation focuses on extracting implicit entities and relationships from clinical narratives and extracting implicit entities from Tweets.

When people use implicit constructs in their daily communication, they assume the existence of a shared knowledge with the audience about the subject being discussed. This shared knowledge helps to decode implicitly conveyed information. For example, the above Twitter user assumed that his/her audience knows that the actress *Sandra Bullock* starred in the movie *Gravity* and it is a movie about space exploration. The clinical professional who wrote the clinical narrative above assumed that the reader knows that *Dolasteron* is an anti-nausea drug. The audience without such domain knowledge may not have correctly decoded the information conveyed in the above examples.

This dissertation demonstrates manifestations of implicit constructs in text, studies their characteristics, and develops a software solution that is capable of extracting implicit information from text. The developed solution starts by acquiring relevant knowledge to solve the implicit information extraction problem. The relevant knowledge includes domain knowledge, contextual knowledge, and linguistic knowledge. The acquired knowledge can take different syntactic forms such as a text snippet, structured knowledge represented in standard knowledge representation languages such as the Resource Description Framework (RDF) or other custom formats. Hence, the acquired knowledge is pre-processed to create models that can be processed by machines. Such models provide the infrastructure to perform implicit information extraction.

This dissertation focuses on three different use cases of implicit information and demonstrates the applicability of the developed solution in these use cases. They are:

- implicit entity linking in clinical narratives,
- implicit entity linking in Twitter,
- implicit relationship extraction from clinical narratives.

The evaluations are conducted on relevant annotated datasets for implicit information and they demonstrate the effectiveness of the developed solution in extracting implicit information from text.

Contents

List of Figures

List of Tables

ACKNOWLEDGEMENTS

The time that I spent at graduate school is a memorable one in life. There are number of people who made it memorable by being with me in both good and difficult times. They have created a great support system which helped me to achieve my goals in the graduate school.

I am greatly thankful for my advisor Dr. Amit P. Sheth. He is the one who saw the potential in me and kept investing resources on me regardless of my failures. He had great instinct on what works for me and guided me on the right direction. He always had the idea on “what is the next thing that I should be looking into” and always connected me with the great minds. His hallway question “what’s new?” kept me thinking about my research. Dr. Sheth always keeps eye on the opportunities for his students. The George Thomas fellowship is a result of that. I was not aware of this prestigious fellowship for graduate students who work on problems related to healthcare issues. But, Dr. Sheth showed me the opportunity and encouraged me to apply for the fellowship and made sure that I get the visibility for my work in the healthcare community. This fellowship opened the doors for few interesting career opportunities. I am specially thankful for the eco-system that he has created for his graduate students. It is an environment that you always have helping hands and creates long lasting bonds with your colleagues. Everyone of us has different expertise and looked at the other’s research with a different perspective. This helps each one of us to get diverse perspectives on our research and think with an open-mind. It is an environment that you don’t have to bother about the administrative work. All of these are taken care for you which makes your life easier and allows to focus on what you are most passionate about. I am really thankful for Dr. Sheth’s hard work on creating this thought provoking environment. Apart from the technical skills that we have, he made sure that we are well-equipped with the soft skills. I came out from every group meeting with the mindset that I have to improve my soft skills, which made me ready for the challenges in the world outside the graduate school. So, thank you Dr. Sheth for all the eye openings, otherwise I could have missed with the busy life at graduate school.

Dr. Prasad is the go-to person at Kno.e.sis. He always had time to listen to me regardless of the topic and corrected my paper submissions at n^{th} time. I had tough time publishing my work on clinical entities and he reviewed each submission ignoring that it was the same content with minor tweaks. His patience, attention to detail, and ability to describe things in simple manner are the qualities that everybody in Kno.e.sis get benefited from. His phrase "when rubber meets the road" always pushed me back to the data and influenced to understand the real problems. Dr. Prasad helped me on improving my writing, he always pointed out the errors in my writing. Many times I wondered how can he handle so many things at once, almost all students in Kno.e.sis go to him to discuss their research problems and he shows equal competency in understanding these problems which are very diverse. This is done in addition to his teaching, multiple proposal submissions, and reviewing project reports. So, thank you Dr. Prasad for all your time you spend on listening to me and helping hand given to improve my research.

My interactions with Dr. Raymer was short. During this short time he understood the difficulty of the problem I tried to solve in my dissertation and quickly started to ask the questions that helped me to understand the breadth of my research question. Pablo Mendes has been a great mentor for last three years of my time in graduate school. I met Pablo later 2013 and we discussed the research problem that I was working on. He was very quick to understand it and provided detailed feedback with great insights. We worked together from there and had fruitful outcomes. He always encouraged me about the merits of my research in the middle of unsuccessful submission efforts and helped to improve my research work until it gets through. Pablo's mentorship was not limited to the technical guidance, he always made sure that my work reached to the right audience. I remember one day he arranged an opportunity for me to present my research work to his group at IBM Almaden which helped me to get the perspective of the people who work in industry on my research topic. In addition to being a mentor, he has been a great friend throughout the years who always wants me to finish my graduate life with great achievements. Thank you Pablo for all your guidance, inspirations and encouraging words.

My first mentor in graduate school was Cory Henson. Cory taught me how to approach a research problem, how to interpret results, and how to separate out the sub-problems. He guided me towards my first publication. Thank you Cory for your guidance when I was naive to the research world. I am grateful for both Meena and Cartic for their great mentorship during my two internships at IBM Almaden. They provided me the opportunity to work with the IBM Watson team at Almaden. Cartic made sure that I get great start with the internship problem by sharing his experience with the research problem as well as helping me with the technical set-up. Meena was instrumental in making sure that I get right problems to work on and provided great insights she had on the domain. They both work hard on making sure that I get the access to the right people at right time during my two Summer internships at IBM. Thank you both for all help you have given me in the past and I am excited to start my career with you two in near future.

Nothing would have been possible if I didn't have a strong support at home. My wife, Sarasi has been always giving me the confidence, supportive and understanding in every hard decisions that we have to make. She always pushes me by asking the question "what are you going to do next in the research?". This kept me progressing on my research topic. It has been a wonderful journey with her at graduate school where we play our researcher roles and at home where we play our family roles and often mixing them together. I am looking forward to the continuation of these mixed roles in years to come. It is always a great pleasure to see how happy and proud my father and sister on my achievements. However, I know they felt alone with my departure to an another country for an unknown time. But, I was never given that feeling, it was only the encouraging words as it was during my stay with them. So, thank you my family for the tremendous support given to achieve my goals.

Graduate school is never fun and productive without the friends I had in and out of the school. I spent good six years of the life with enthusiastic colleagues who all are thriving to achieve the similar goals as mine. In some instances we failed and in some instances we earned the success with lot of hard work and persistence. Ultimately, we successfully completed the first step of achieving our goals with a great support from each other in different ways. I would like to thank Ajith Ranabahu. Ajith helped to start the life in USA. His advices on matters both in and out of the school helped for a smooth transition from life in Sri Lanka to life in USA. My colleagues in Kno.e.sis were very constructive in giving the feedback on my research work and to my writing, which always improved the quality of my work. The coffee and lunch breaks with

Pavan Kapanipathi, Revathy Krishnamurthy, Gaurish Anand, Siva Kumar Ch, Adarsh Alex, and Vaikunth Sridharan were always about controversial topics, which added fuel to the working day. Listening to Sanjaya Wijeratna's long stories improved my patience. The focus that Lu Chen and Wenbo Wang had in the lab, persistence of Vinh Nguyen with her research, multi-tasking ability of Pramod Anantharam, calmness of Raghava Mutharaju, Pavan Kapanipathi's effort to give quality feedback to other's work are few things that I learnt from my peers. So thank you all past and present Kno.e.sis members for these lifetime memories and lessons. Thank you Tonya Davis for taking care of all administrative things for me. You have been helpful throughout last six years. Jibril Ikharo helped me by proofreading my writing. Thank you Jibril for your time. I thank the Sri Lankan community in Dayton who kept the life in Dayton eventful.

Finally, I would like to thank funding sources that enable the research in this dissertation. Wright State computer science department gave me the opportunity to work as a teaching assistant, ezDI funded my research work on clinical data, Health data consortium funded my research through George Thomas graduate fellowship, and my work on Twitter data is funded by NSF grant CNS 1513721 titled 'Context-aware Harassment Detection on Social Media.'

Dedicated to

My father and wife

1

Introduction

People communicate their ideas, opinions, and facts using natural language. It is one of the powerful tools that we as humans collectively developed over hundreds of thousands of years and will continue to develop in years to come. While the debate on the evolution of the languages has not achieved a consensus, the theories have estimated that it first evolved around 150,000 - 350,000 years ago, which is roughly the time frame accepted for the evolution of modern Homo sapiens [Origin of language 2004]. English is one of the most commonly spoken language today has evolved over 1,400 years on its own [English language 2001]. The evolution of language is influenced by social, cultural, and economical changes which have taken place in the society. For example, the industrial revolution took place in the 18^{th} and 19^{th} century and had a great impact on these three dimensions. As a consequence, it had a great influence on the evolution of the English language [Bragg 2006]. One recent event that is impacting the evolution of the language is the rapid progress of technology. It is changing the vocabulary, adding new language constructs, and changing the style of communication. For example, the verb ‘to google’ was added to the English language to mean the activity of searching the Web, and the known term ‘text’, which previously referred to any written work, is now also being used to mean the activity of sending short messages over mobile networks. The rapid progress of technology has also yielded new communication media. Social media platforms that have emerged in recent years are examples of such media. Twitter, Facebook, and Google+ are a few social media platforms have become popular in recent years. People communicate effectively on these platforms with short messages. Studies have shown that the effective length of a message on these platforms is between 40 to 100 characters [Lee 2014] [Kolowich 2016]. In fact Twitter allows only 140 characters for composing a message (a message on Twitter is commonly referred to as a *tweet*). Such studies, restrictions imposed by the communication platforms, assumptions about shared understanding, and creativity of the human beings have added features to the language usage.

One such feature added to the language is the ability to express ideas, opinions, and facts in an *implicit* manner. According to the Merriam-Webster dictionary[1], the term *implicit* means 'capable of being understood from something else though unexpressed.' Consider the tweet, *'I'm striving to be positive in what I say on Twitter. So I'll refrain from making a comment about the latest Michael Bay movie.'* The tweet has a negative opinion on the movie *Transformers: Age of Extinction* (a conclusion founded on the basis of the facts that this tweet was sent on July 2014, and Michael Bay released *Transformers: Age of Extinction* in June 2014). The sentence *'the respirations were not labored and there were no use of accessory muscles'* in an electronic medical record (EMR) states that the patient does not have the medical condition *shortness of breath*. However, neither the negative opinion nor the movie is explicitly mentioned in the tweet and the *shortness of breath* is not explicitly mentioned in the sentence extracted from the EMR. The language skills that humans possess and the available knowledge about the domain allow one to decode implicitly conveyed information. Extracting such implicit information has not gained the attention by computer scientists. This dissertation argues that careful usage of domain and contextual knowledge with an ability to bridge the semantic gaps of language usage can address the problem of automatically extracting implicit information from unstructured text.

[1] http://www.merriam-webster.com/

1.1 Why People Use Implicit Constructs in Communication?

The above tweet contains an element of sarcasm; it is observed that people heavily use implicit constructs to express sarcasm [Davidov et al. 2010]. However, expressing sarcasm is not the only instance where people tend to use implicit constructs. It is also used in instances such as: 1) When trying to recall forgotten things; for example, someone trying to recall the movie *Gravity* would say *'the space movie with Sandra Bullock'*. 2) To provide descriptive information; for example, it is a common practice to describe the features of an entity rather than simply list down its name in clinical narratives. Consider two sentences *'extensive fluid accumulation within the abdominal cavity predominantly anterior to the stomach and in the anterior pararenal space'* and *'small fluid adjacent to the gallbladder with gallstones which may represent inflammation'*. The first sentence contains implicit mention of the condition *edema* while the second sentence contains implicit mention of the condition *cholecystitis*. Both sentences provide important information about the patient's health status that would be missing if they choose to list only entity names. The condition *edema* means 'fluid accumulation in body tissues or cavities', hence simple mention of term 'edema' would provide incomplete information about the body location. *Cholecystitis* means 'inflammation in gallbladder' and it may result from a variety of factors. The second sentence provides a detailed explanation of *cholecystitis* with the possible cause. While it is feasible to provide these extra information with the corresponding explicit entity names, it is observed that clinical professionals prefer this style of writing with implicit information as they are often typed during a patient visit in a way that is natural in spoken language and meant for consumption by the professionals with similar backgrounds. The extra information in description can be critical in understanding the patient's health status and treating the patient. 3) To emphasize the features of an entity; for example, the tweet *'Mason Evans 12 year long shoot won big in golden globe'* has an implicit mention of the movie *Boyhood*, but the speaker wants to emphasize its long shooting time and its awards rather than the movie itself. 4) To communicate shared understanding; for example, medical documents rarely mention the relationship between entities (e.g., between symptoms and disorders, between medications and disorders), rather it is understood that the other professionals reading the document have the necessary expertise to understand such implicit relationships in the document. These examples illustrate some common reasons to use implicit constructs and their value in day to day communication.

1.2 The Significance

Identification and extraction of implicit information from unstructured text has not gained the attention it deserves. Implicit constructs are a common occurrence in text. An analysis with 300 electronic medical records of cardiovascular patients showed that 35% of *edema* mentions and 40% of *shortness of breath* mentions are expressed in an implicit manner (*edema* and *shortness of breath* are common symptoms among cardiovascular patients). A similar analysis on a tweet corpus found that 20% of movie mentions and 40% of book mentions are stated in an implicit manner. The relationships between entities in clinical documents (e.g., symptom *s* caused by disease *d*) are rarely mentioned explicitly. Given the volume of implicit constructs in text, failure to identify and extract them can result in significant loss to the applications that depend on such data sources as demonstrated in following examples.

- Clinical documents play a central role in understanding a patient's health status and improving the quality of delivered care. It is a common practice to spot the clinical conditions mentioned in these documents and assign them unique identifiers (e.g., ICD9 or ICD10). These identifiers are important for unambiguously representing the medical conditions, for preparing the post-discharge plan, for maintaining patient's history, and for performing secondary data analysis tasks. The sentence *'Patient is comfortably breathing in room air but shows re-accumulation of fluid in extremities'* has implicit mentions of the clinical conditions *shortness of breath* and *edema*. A medical professional reading this sentence would identify the mentions of these two conditions (the first negative and the second positive) and assign them the required identifiers accordingly, but applications developed to process clinical notes fail to identify such implicit mentions. Failure to identify such mentions may significantly affect completeness of the information captured about the patients and, ultimately, the quality of the care delivered.

- Tweets have been a popular data source for applications like opinion analysis, trend detection, and event

Application	**Tweet**	**Implicit Entity**
Sentiment Analysis	*'New Sandra Bullock astronaut lost in space movie looks absolutely terrifying'*	*Gravity*
Trend Detection	*'Kinda sad to hear about that South African runner kill his girlfriend'*	*Oscar Pistorius*
Event Monitoring	*'Texas Town Pushes for Marijuana Legalization to Combat Cartel Traffic'*	*El Paso*

Table 1.1: Examples of tweets with implicit mentions that are valuable but often missed in current applications that rely on explicit mentions only

monitoring. Each of these applications depends on identifying the entities mentioned in tweets. Failure to identify the entities mentioned in tweets negatively affects the outcome of the application. Table 1.1 shows the example tweets with their implicit entities and the related application for each tweet. A sentiment analysis tool wouldn't identify that the positive sentiment is expressed towards the movie *Gravity* in the first tweet unless it is provided with the mechanism to decode implicit entities in the text. A trend detection application wouldn't detect that *Oscar Pistorius* is trending if it is not able to identify the references to him in the tweets like the second tweet in Table 1.1, as he was frequently referred to with similar phrases in tweets. An application that monitors opinion on marijuana legalization in different states in the United States wouldn't identify that the third tweet is relevant to marijuana legalization in *El Paso* unless it could decode implicit mentions of locations in tweets.

1.3 Focus of the Dissertation

The term 'implicit' is defined as 'understood though not clearly or directly stated'.[2] The task of implicit information extraction can be described as automatically extracting implied but not plainly expressed structured information from unstructured text. While it is possible to express any information component (e.g.,

[2] http://www.merriam-webster.com/dictionary/implicit

entities, relationships between concepts, sentiment, opinion, and emotions) in an implicit manner, this dissertation is focused on extracting implicit entities and implicit relationships between clinical entities. The broader definition of the term entity is that it has a distinct existence as an individual unit.[3] In the literature, the unambiguous terminal pages in Wikipedia are frequently used as entities [Guo et al. 2013]. Since the focus of this dissertation is the task of entity linking and the objective of entity linking is to determine the identity of the entities in the text, we consider anything that is defined in a referent knowledge resource with an unique identity as an entity. A referent knowledge resource can be a dictionary, a thesaurus, a structured knowledge base, or a vocabulary (E.g., resources defined with unique URLs in linked open data or resources defined with unique IDs in Unified Medical Language System (UMLS) are considered as entities). This dissertation defines an implicit entity as an entity mentioned in the text where neither its name is explicitly present nor it is a synonym/alias/abbreviation of an entity name or a co-reference of an explicitly mentioned entity in the same text.

Entities appear in any type of text. Identifying their mentions and their type is referred to as named entity recognition. Finding the identity of the mentioned entities is referred to as entity linking. These tasks in different text types pose different challenges. Techniques developed to link the explicit mentions of entities in more organized text like news, Wikipedia, and clinical documents fall short in linking entities that appear in less organized text like tweets and short text messages (SMS) [Derczynski et al. 2015] [Ferragina and Scaiella 2010]. This is due to the fact that the former text is more grammatical, structured, lengthy, and provides rich context, while the latter text is short, conversational, ungrammatical, noisy, and contains unorthodox abbreviations, whose interpretation depends on the context. This dissertation focuses on solving the implicit entity linking problem in both types of texts. To demonstrate the different challenges posed by each text type and to demonstrate the effectiveness of the proposed solution, it uses clinical documents and tweets as two distinct data sources.

Clinical documents are rich in information content and contain patient's clinical history, family history, current diagnoses, current medications, test results, etc. The entities in clinical documents can be diseases,

[3] https://en.wiktionary.org/wiki/entity

symptoms, medications, procedures, etc. The diseases and symptoms are more likely to be mentioned in an implicit manner than other entities due to the necessity of making distinction between (abstract) clinical observations as described above and the (detailed) descriptive explanation of observations required for insights into the patient's health status. Hence, this dissertation focuses on linking implicit mentions of diseases and symptoms as a use case.

Tweeting is a popular mechanism to discuss any topic ranging from what people have done during the course of the day, opinions about new movie, and climate change on earth to presidential elections in the USA. Hence, tweets may contain mentions of any entity type. To demonstrate the effectiveness of the implicit entity linking techniques in tweets, this dissertation focuses on entities of the type movie and book, since they offer an agreeable level of difficulty for human annotators.

Implicit relationships are a common occurrence in clinical documents. Clinical documents contain diseases and symptoms separately, but do not explicitly state that a certain disease causes a certain symptom. A medical professional reading the document can connect all such occurrences and form the relationships which provide valuable insight into the patients' health status. Such relationships also exist between medications and diseases/symptoms, and procedures and diseases/symptoms. This dissertation uses the extraction of implicit relationships between diseases and symptoms as the use case to demonstrate the effectiveness of the developed algorithm.

It is worth noting the difference between the terms 'implicit' and 'inference'. The former deals with decoding information expressed indirectly and can be understood from something else that acts as a clue. The latter involves deriving new facts from evidence by (logical) reasoning, typically by applying abductive or deductive reasoning techniques. For instance, consider two sentences *'small fluid adjacent to the gallbladder with gallstones'* and *'small fluid adjacent to the gallbladder with gallstones which may represent inflammation'*. The first example discusses only the presence of gallstones in gallbladder, and a medical professional would *infer* that the patient may have *cholecystitis* based on this information. But sentence itself does not state the presence of *cholecystitis*. However, the second sentence has *implicit* mention of entity *cholecystitis* since it has extra phrase indicating 'inflammation in gallbladder'. This dissertation focuses on the latter, i.e., it does not aim to derive new facts based on formal reasoning, rather it elicits indirectly mentioned facts in the text. The task of implicit information extraction is significantly different from generating new facts using inference.

1.4 Challenges

The implicit information extraction poses different challenges from extracting explicit information. For example, extracting implicit entity mentions requires detecting and disambiguating clues to uncover entity mentions as they do not contain entity names. The absence of the entity names is redressed/remedied by leveraging different characteristics of the entities. The tweet *'I'm striving to be positive in what I say on Twitter. So I'll refrain from making a comment about the latest Michael Bay movie'* has implicit mention of the movie *Transformers: Age of Extinction* and it uses the director to indicate the movie while the tweet *'Mason Evans 12 year long shoot won big in golden globe'* uses a unique feature of the movie and one of its characters to indicate the implicit reference to the movie *Boyhood*. The typical entity linking solutions depend on the presence of the entity names to identify the presence of entities and link them to the correct entries in the knowledge base. Hence, the absence of the entity names poses different challenges in linking implicit entities than the explicit entities. As mentioned earlier, it needs to recognize the clues in the text that may indicate a mention of an entity and use them to identify the correct entry in the knowledge base.

The characteristics of the entities that may be used to indicate the entities implicitly can be very diverse. Consider the following phrases used to refer to the movie *Boyhood*:

- 'Richard Linklater movie'
- 'Ellar Coltrane on his 12-year movie role'
- '12-year long movie shoot'
- 'latest movie shot in my city Houston'
- 'Mason Evan's childhood movie.'

The first two phrases refer to the movie through its director and actor, the third phrase uses a distinctive feature (it was shot over a long period), the fourth example uses the shooting location of the movie, and the last one refers to it with a character in the movie. This shows the vast extent of the knowledge about the entity that an implicit entity linking solution should possess and exploit in order to perform implicit entity linking. Acquiring such a comprehensive domain knowledge is a challenge that should be addressed by any implicit entity linking solutions.

Another unique challenge in interpreting implicit mentions of entities is the acquiring temporally relevant knowledge. An identical phrase referring to a particular entity during the time period t_1 may refer to a different entity during another time period t_2 – e.g., the phrase ‘space movie’ may refer to the movie *Gravity* in Fall 2013, while it may refer to the movie *The Martian* in Fall 2015. On the flip side, the most salient characteristics of the movies may change over time, and so will the phrases used to refer to them. The movie *Furious 7* was frequently referred to with the phrase *‘Paul Walker’s last movie’* in November 2014. This was due to the actors’ passing around that time. However, after the movie release in April 2015, the same entity was often mentioned using the phrase ‘fastest film to reach the $1 billion.’ Hence, the solutions developed to solve the implicit entity linking should possess timely-relevant domain knowledge.

The manifestation of explicit and implicit entities in the text exhibit syntactic differences. The explicit entity mentions are signified by the contiguous noun phrases. However, the implicit entity mentions are not necessarily signified by the contiguous text fragments nor are they always made of noun phrases. Consider the example, *‘when Augustus tells Hazel his cancer is back is the point where i lose my shit for the rest of the movie.’* It is understood that the two phrases *‘Augustus tells Hazel’* and *‘movie’* are the minimum text fragments that one would need to determine that it has a mention of the movie *The Fault in Our Stars*. These phrases are neither contiguous nor do they only contain noun phrases, as shown by the following output of the part-of-speech tagger.

Augustus/NNP tells/VBZ Hazel/NNP ... movie/NN

This shows the syntactic challenges involved in resolving implicit entity mentions.

Another challenge involved in the resolution of implicit entities is the handling of embedded negations.

This is a prominent feature in clinical text. Negation detection is traditionally considered as a separate task from the entity linking. This separation is feasible with explicit entity mentions since a term indicating negation is separable from the entity mention, whereas in implicit mentions, it is often embedded into the phrases with the entity mentions (e.g., *'patient denies shortness of breath'* vs *'patient is breathing comfortably'*). Hence, negation detection is considered a sub-task of implicit entity linking. Negation detection of implicit entities requires one to understand the phrases that convey the opposite of characteristics exhibited by the entity, which can be more involved than knowing the antonyms.

Similar characteristics are also exhibited by implicit relationship manifestations in text. A typical solution that is designed to extract causal relationships from text would look for the presence of terms like 'cause', 'induce', 'lead to', and 'give rise to'. However, implicit mentions of such relationships do not contain these terms. It again requires comprehensive domain knowledge about the relationship between entities to establish such relationships.

These features shown by the text with implicit information warrant new solutions to extract implicit information from the text.

1.5 Role of the Domain Knowledge

A human reading a text with implicit information would be able to decode implicit information only if he/she has relevant knowledge of the domain. A reader who does not know that *Sandra Bullock* starred in the movie *Gravity* and that it was about space exploration would not have identified the mention of the movie *Gravity* in the first tweet of Table 1.1. A reader with no knowledge of the shooting incident which involved *Oscar Pistorius* would not have identified his mention in the second tweet of Table 1.1. A reader with no medical knowledge would not have any clue of the relationship indicated between *nausea* and *dolasteron* in *'He is suffering from nausea and severe headaches. Dolasteron was prescribed.'* Hence, one of the main ingredients of a solution aimed at solving the implicit information extraction problem is the use of relevant domain knowledge. Chapter 2 of this dissertation discusses the importance of different types of knowledge in extracting implicit information from the text in detail.

1.6 Implicit Information Extraction

This dissertation demonstrates a framework consisting of four main components in order to solve the problem of implicit information extraction from the text. These components perform following tasks:

- **Knowledge Acquisition**: Knowledge plays a major role in extracting implicit information from the text. The required knowledge for each task exists in different sources in different formats. Some of them are well-known knowledge sources that organize and represent knowledge using standard methodologies while other sources are not designed to serve as structured knowledge sources. Hence, the first task of the framework involves identifying and extracting the relevant knowledge from the organized knowledge sources and develop techniques to extract relevant knowledge from other non-organized knowledge sources. It also develops techniques to acquire additional knowledge when the existing knowledge in identified sources is insufficient.

- **Knowledge Modelling**: Modelling the acquired knowledge in a machine readable manner is an important step in extracting implicit information. The modelled knowledge may represent entities of interest and their hierarchical and non-hierarchical relationships with associated metadata.

- **Detecting Implicit Information**: Detection of possible implicit mention of information in the text is the first step to extract them. The framework consists of components that can detect the presence of implicit entities in clinical narratives and tweets, and implicit relationships in clinical documents. This component identifies the semantic cues that may indicate the presence of implicit information in the text.

- **Information Extraction**: The first two components build the infrastructure to extract and match implicit information from the text and the third component finds the text that may contain implicit information. The information extraction component performs the final goal of the framework, namely, implicit entity linking and implicit relationship extraction. In order to achieve this goal, this component deploys supervised techniques, unsupervised techniques, and semi-supervised techniques in tasks of implicit entity linking in tweets, implicit entity linking in clinical narratives, and implicit relationship extraction in clinical narratives respectively.

1.7 Contributions of the Dissertation

This dissertation makes several contributions:

- Identify and demonstrate the value of implicit information.
- Study the characteristics of the implicit information manifestations.
- Demonstrate the value of knowledge in extracting implicit information.
- Develop a framework for implicit information extraction and demonstrate its usage in three implicit information extraction problems.

1.8 Dissertation Organization

The rest of the dissertation consists of five chapters. Second chapter discusses the role of knowledge in natural language understanding tasks. While this dissertation focuses on extracting implicit entities and relationships, the second chapter discusses the requirement of knowledge for decoding different types of implicit information. Then it survey the attempts taken to serialize the knowledge in different formats by the artificial intelligence community. Specifically, it discusses resources that contain linguistic knowledge and world knowledge and how such sources can help in decoding implicit information in the text.

The next three chapters demonstrate how the described implicit information extraction framework is realized for performing three distinct information extraction tasks. Each chapter describes the problem at hand with examples and demonstrates the characteristics of the implicit information manifestation. This includes implicit entity manifestations in clinical narratives and tweets, and implicit relationship manifestations in clinical narratives. Once the problem is identified and demonstrated with examples, each chapter describes how the four components of the framework are realized, namely, knowledge acquisition, knowledge modelling, detecting Implicit Information, and information extraction. This includes identifying relevant knowledge sources. The examples demonstrated in each chapter provide hints for the nature of the knowledge sources needed. These sources include structured/unstructured and domain-specific/cross-domain knowledge sources. Then each chapter demonstrates how the acquired knowledge is processed to create knowledge models. This process is different for each chapter due to the nature of the knowledge source that it deals with. Similarly, the information extraction step of each chapter develops techniques that suite the knowledge models and the manifestations of implicit information. After demonstrating the realization of the framework, each chapter presents the results of the evaluation conducted.

The sixth chapter summarizes the findings and concludes the dissertation by identifying interesting future directions.

2

The Role of Knowledge in Implicit Information Extraction

Understanding a natural language sentence requires knowledge [Winograd 1972] [Chandrasekaran et al. 1999] [Ovchinnikova 2012]. According to Merriam-Webster dictionary,[1] knowledge is defined as "facts, information, and skills acquired by a person through experience or education; the theoretical or practical understanding of a subject." The required knowledge in understanding natural language text consists of knowledge of the grammar of a language, knowledge about the meanings of words in a language, knowledge of the context of a sentence, knowledge about the world, and common sense knowledge. These knowledge components, which a human gradually builds by learning over his/her lifetime, seamlessly integrate and get used in language interpretation. Inspired by this observation, algorithms developed for performing natural language understanding tasks attempt to exploit the resources that contain these knowledge components. Such resources have been exploited for natural language understanding tasks, including word sense disambiguation [Banerjee and Pedersen 2002] [Li et al. 1995], co-reference resolution [Ponzetto and Strube 2006] [Rahman and Ng 2011], entity disambiguation [Mendes et al. 2011] [Cucerzan 2007] [Bunescu and Pasca 2006], and question answering [Hsu et al. 2008] [Ferrucci et al. 2010]. This chapter discusses the role of knowledge in

[1] http://www.merriam-webster.com/dictionary/knowledge

the implicit information extraction task.

By definition, implicit information is not spelled out in the text. The speaker assumes a shared knowledge with the audience on the topic being discussed. This shared knowledge helps the audience to decode the things that are not explicit, but still necessary to understand the message communicated. The shared knowledge can be of different types. This dissertation will discuss different types of shared knowledge needed to decode implicit information using the examples in Table 2.1.

- The first example in the Table 2.1 is a tweet and it has an implicit mention of the movie *Gravity*. The fact that it talks about a movie with astronauts and mention of Sandra, which refers to the actress *Sandra Bullock*, are the clues that help to decode the implicit mention of the movie *Gravity*. The speaker assumes that the audience knows that the movie *Gravity* is related to space exploration and *Sandra Bullock* starred in it.

- The second example is also a tweet and it has an implicit mention of the movie *Gravity*. This was tweeted around the time of the Mars Orbiter Mission conducted by the Indian Space Research Organization (ISRO).[2] The tweet compares the budget of the Mars Orbiter Mission and the movie *Gravity*. It was found that the former has a lower budget than the latter; the general public found this fact interesting and started to talk about this topic in social media. The readers of this tweet are expected to know about the Mars Orbiter Mission and the fact that it had lower budget than the movie *Gravity* in order to spot the implicit mention of the movie *Gravity* in this tweet.

- The third example is extracted from a clinical narrative and it contains implicit mentions of the medical conditions *edema* and *shortness of breath*. The medical condition *shortness of breath* is clinically defined as 'uncomfortable sensation of breathing' or 'labored breathing' and the medical condition *edema* is clinically defined as an 'accumulation of an excessive amount of watery fluid in cells or intercellular tissues.'[3] According to these definitions, the sentence contains a positive mention of *edema* and a negative mention of *shortness of breath*. The reader would need to know about the physiological observations

[2] `https://en.wikipedia.org/wiki/Mars_Orbiter_Mission`

[3] These definitions are obtained from unified medical language system (UMLS)[Bodenreider 2004].

	Example	Implicit Information
1	*'It's hard for me to imagine movie stars as astronauts, but the movie looks great! and who doesn't like Sandra.'*	Gravity
2	*'ISRO sends probe to Mars for less money than it takes Hollywood to send a woman to space.'*	Gravity
3	*'The patient showed accumulation of fluid in his extremities, but respirations were unlabored and there were no use of accessory muscles.'*	Shortness of breath and edema
4	*'He is suffering from nausea and severe headaches. Dolasteron was prescribed.'*	Dolasteron prescribed for nausea
5	***Andy** : Shall we meet noon tomorrow?* ***Bob** : I have a lunch-out with Sarah*	declined the invitation
6	*'John fell, Max pushed him.'*	temporal and causal relationships

Table 2.1: Few examples with implicit information. The first two tweets contain implicit mentions of the movie *Gravity*, third sentence is from a clinical note and contains implicit mentions of clinical entities *shortness of breath* and *edema*, fourth example contains implicit relationship 'dolasteron is prescribed for nausea'. Last two examples are snippets from daily communications. The fifth example implicitly declines the invitation to meet and last example has implicit causal and temporal relationships.

of clinical conditions and have knowledge about the meaning of the words in the English language in order to decode the positive and negative implicit mentions of these medical conditions in this example.

- The fourth example, again, is an extracted sentence from a clinical narrative. This example has stated a relationship in an implicit manner. *Dolasteron* is an anti-nausea drug. The example says the patient has *nausea* and was prescribed *dolasteron*, but never explicitly says that *dolasteron* was prescribed for *nausea*. The reader is expected to know the relationship between *nausea* and *dolasteron* in order to decode the relationship stated in an implicit manner.

- The fifth example is a dialog between Andy and Bob. In the example, Bob never declined Andy's invitation to the lunch explicitly. However, Andy could conclude that it is not possible to meet Bob at the noon the following day. This conclusion requires Andy knowing that the event lunch-out with Sarah will take place at noon in a different place and that a person generally cannot participate in two events taking place in different locations at the same time.

- Example 6: The final example was discussed in [Ovchinnikova 2012]. There is a temporal and causal relationship between the two incidents discussed in this example. Temporally, the incident 'John fell' happens after the incident 'Max pushed him' and the latter caused the former to happen. The knowledge like "when something is pushed, normally, it moves" helps to establish these implicit relationships [Ovchinnikova 2012].

These examples show the important role played by the knowledge possessed by humans in decoding the implicit information in the text. The knowledge being used for natural language understanding tasks is generally of two types: 1) linguistic knowledge, and 2) world knowledge. The rest of this chapter will discuss the difference between these knowledge sources and their applicability to the above examples.

2.1 Linguistic Knowledge

Linguistic knowledge is focused on knowledge about a language, and it aims to represent the lexical meaning of the words in a language. There are several approaches proposed in the literature to implement this idea. Katz and Fodor proposed to capture meanings using *semantic markers* [Katz and Fodor 1963]. Fillmore [Fillmore 1976], Jackendoff [Jackendoff 1987], and Dowty [Dowty 1991] proposed to decompose the meanings of the verbs into thematic roles. Other proposed approaches include selectional preferences [Chomsky 1964] [McCawley 1976], prototype theory [Roach et al. 1978], and generative lexicon [Pustejovsky 1991]. An approach which is different than the ones noted above is that of lexical-semantic relations. With this approach, the meanings of words are represented as a network of relationships between word senses [Cruse 1986].

The most popular artifacts of the approaches proposed above are FrameNet [Ruppenhofer et al. 2006] and WordNet [Miller et al. 1990] [Miller and Fellbaum 1991] [Fellbaum 1998]. FrameNet has been developed by realizing the ideas of frame semantics proposed by Fillmore and WordNet has been developed by realizing the idea of lexical-semantic relationships. These two resources are frequently used by modern natural language understanding applications.

2.1.1 FrameNet

FrameNet [Ruppenhofer et al. 2006] is based on frame semantics. The basic idea is that the meanings of most words can be understood by semantic frames. Semantic frames consist of a description of the type of event, relation or entity it is associated with and the participants in it. For example, the concept of cooking typically involves a person doing the cooking (Cook), the food that is to be cooked (Food), something to hold the food while cooking (Container), and a source of heat (Heating_instrument).[4] The FrameNet defines a frame called 'Apply_heat' and the elements described above will become its frame elements. Such frames help to understand the components of the sentences and their roles.

VerbNet [Kipper et al. 2000] and PropBank [Martha et al. 2005] follow the same ideas to represent the

[4] https://framenet.icsi.berkeley.edu/fndrupal/about

Word Type	**Total**
Noun	5513
Verb	5172
Adjectives	2385
Other POS	472
Total Frames	1223
Total Frame Relations	1841

Table 2.2: Statistics of the FrameNet

meanings of the words. However, FrameNet has more things to offer than VerbNet and PropBank. VerbNet describes only the semantics of verbs and PropBank is restricted to the verbs which appear in Penn TreeBank. FrameNet describes the frame semantics of verbs, nouns, adjectives, adverbs, and prepositions. Another unique feature offered by FrameNet is the relationship between semantic frames. This is a very useful feature in reasoning over natural language sentences. For example, there is a causation relationship defined between the semantic frames of KILLING and DEATH, and the semantic frame DEATH contains Is_Preceeded_by relationship with the BEING_BORN and GIVING_BIRTH frames. These relationships allow researchers to extract causative and temporal information from natural language sentences which are otherwise implicit in the text (e.g., Example 6 in Table 2.1). Table 2.2 shows the latest statistics from FrameNet. The 1,841 relationships mentioned in the table that are defined between semantic frames are of the types: 1) Is_Inherited_by, 2) Uses, Subframe_of, 3) Is_Preceded_by, 4) Perspective_on, 5) Is_Inchoative_of, 6) Is_Causative_of, and 7) See_also. Table 2.3 demonstrates examples for each relation.[5]

2.1.2 WordNet

WordNet is the most popular lexical-semantic knowledge base in the natural language understanding literature. This is due to the fact that it has huge lexical and conceptual coverage, and is available in multiple

[5] These examples are found in [Ovchinnikova 2012].

Relationship	**Example**
Inheritance	GIVING - COMMERCE_SELL
Causative_of	KILLING - DEATH
Inchoative_of	COMING_TO_BE - EXISTENCE
Perspective	OPERATE_VEHICLE - USE_VEHICLE
Precedence	FALL_ASLEEP - SLEEP
Subframe	SENTENCING - CRIMINAL_PROCESS
Uses	OPERATE_VEHICLE - MOTION
See_also	LIGHT_MOVEMENT - LOCATION_OF_LIGHT

Table 2.3: Examples for each relationship defined between semantic frames in FrameNet

languages. The design principles of WordNet are inspired by psycholinguistic and computational theories of human lexical memory [Miller and Fellbaum 1998]. WordNet has a concept called *synsets* that groups words that have the same sense. Each synset represents an underlying lexical concept and it is assigned a definition referred to as 'gloss'. WordNet contains synsets prepared for nouns, verbs, adjectives, and adverbs. These synsets are linked using relationships like hyponymy, meronymy, and antonymy. Hyponymy relation signifies the super-subordinate relationship between synsets. For example, armchair is an hyponym of chair and chair is a hyponym of furniture. Meronymy signifies the part-whole relationship such as leg is a part of chair. The antonymy relationship is used to link synsets with opposite meanings. In addition to such semantic relationships, WordNet contains relationships that establish the connection between syntactic variations of the same words. Adjectives are linked to the nouns that they are derived from using a relationship called 'pertainyms'; the same relationship is used to establish the link between adjectives and adverbs. Table 2.4 shows the statistics of the WordNet 3.0.

The language skills possessed by humans help them to identify implicit information in text. Assuming that one knows the definitions of the clinical conditions *edema* and *shortness of breath*, he/she still should know that the term 'unlabored' is an antonym of the term 'labored', the term 'respiration' is a synonym

POS	Unique Strings	Synsets
Noun	117798	82115
Verb	11529	13767
Adjective	21479	18156
Adverb	4481	3621

Table 2.4: Statistics of WordNet 3.0

of 'breathing', and the phrase 'watery fluid' has the same sense as the term 'fluid' to understand that the third example in Table 2.1 has an implicit positive mention of the clinical condition *edema* and an implicit negative mention of the clinical condition *shortness of breath*. Semantic and syntactic relationships described in lexical-semantic knowledge bases like WordNet can be leveraged to provide such knowledge required by algorithms to perform implicit information extraction in natural language text as this dissertation will demonstrate with the task of linking implicit entities in clinical narratives.

2.2 World Knowledge

World knowledge includes knowledge about a particular subject (e.g., medicine), general knowledge about the world, or common sense knowledge. The interest in representing world knowledge started decades ago. The researchers working on Artificial Intelligence (AI) realized that capturing world knowledge would be an important step towards advancing AI solutions. They referred to these models as *ontologies*. Tom Gruber defined an ontology as "an explicit specification of a conceptualization" [Gruber 1993]. Ontologies developed over the years representing world knowledge cover, knowledge on specific subjects (e.g., the Gene Ontology [Ashburner et al. 2000] and Unified Medical Language System [Bodenreider 2004] cover the gene and clinical domains respectively), generic knowledge (e.g., DBpedia [Lehmann et al. 2014] which was created by extracting structured data from Wikipedia), and common sense knowledge (e.g., the OpenCyc ontology[6] and the ConceptNet ontology [Liu and Singh 2004]). Nowadays, representing world knowledge in a structured

[6] http://www.opencyc.org

format has become a very popular task.

The advent of Semantic Web technologies has given a boost to the ontology development task. These ontologies primarily use Resource Description Framework Schema (RDFS)[7] and Web Ontology Language (OWL)[8] as the knowledge representation languages. In addition, Semantic Web technology has developed a diverse set of tools that help to: 1) develop ontologies[9], 2) reason over them[10], and 3) visualize the knowledge represented with ontologies[11]. Perhaps the most useful artifact of the Semantic Web technology is the Linked Open Data (LOD) cloud[12]. LOD contains ontologies developed by the Semantic Web community. These ontologies are interlinked to represent the relationships between the concepts present in different ontologies. The interlinking of the concepts allows the exploration of the knowledge represented about the same concept in different knowledge bases (possibly in different contexts; e.g., one ontology may represent *Barack Obama* as president of the United States while another ontology may represent knowledge associated with him as an author). The knowledge represented in LOD has been used by many natural language understanding tasks.

Knowledge about the world plays an important role in understanding implicitly stated facts in natural language text. A reader who does not know that *Sandra Bullock* starred in the movie *Gravity* and it is a movie about space exploration would have no clue that the first example in Table 2.1 has a mention of the movie *Gravity*; a reader who does not know about the Mars Orbiter Mission and the fact that it had a lower budget than Hollywood movie *Gravity* would not be able to decipher the mention of movie *Gravity* in the second example in Table 2.1; and a reader who does not know the characteristics of the clinical conditions *shortness of breath* and *edema* would not be able to decode their mentions in the third example in Table 2.1. Similarly, a reader would need to know that *dolasteron* is a medication prescribed for the clinical condition *nausea* in order to successfully decode the implicit relationship stated in the fourth example in Table 2.1.

[7] `https://www.w3.org/TR/rdf-schema`

[8] `https://www.w3.org/OWL`

[9] `http://protege.stanford.edu`

[10] `http://owl.cs.manchester.ac.uk/tools/list-of-reasoners`

[11] for example see `http://vowl.visualdataweb.org/webvowl.html`

[12] `http://lod-cloud.net`

2.3 Conclusion

This chapter showed the importance of knowledge in decoding implicitly stated facts in natural language text. It started with examples of natural language text that contain implicit information. Then it discussed both linguistic and world knowledge available to process these statements and how they might help in decoding implicit information stated in the text. It is clear that knowledge plays a critical role in deciphering implicit information in text. This motivated the research presented in this dissertation to explore the knowledge available in different forms for decoding implicit information in text. It shows how to use the clinical knowledge available in UMLS and the linguistic knowledge available in WordNet to identify the entities and causal relationships mentioned implicitly in clinical narratives, and how to use world knowledge available in generic knowledge bases in identifying entities mentioned implicitly in tweets.

3

Implicit Entity Linking in Clinical Documents

3.1 Overview

Consider the sentence: *'Patient has shortness of breath with reaccumulation of fluid in extremities'*. This is a sentence extracted from a clinical narrative and it states that the patient has the clinical conditions *shortness of breath* and *edema*. The former is explicitly mentioned, while the latter is implied by the phrase *'reaccumulation of fluid in extremities'*. Implicit entity mentions are a common occurrence in clinical documents as they are often typed during a patient visit in a way that is natural in spoken language and meant for consumption by professionals with similar backgrounds. Table 3.1 contains more such example sentences with implicit entity mentions.

Linking implicit mentions of the entities in clinical narratives is an important task as they are critical information which is required by downstream applications such as preparing post-discharge plans, performing secondary data analysis tasks, and preparing billing documentation. A medical professional reading the sentences listed in first column of Table 3.1 would identify the mentions of the entities listed in the corresponding row of the second column. However, the solutions developed to perform entity linking in clinical documents

Text	**Implicit Entity**
'Rounded calcific density in right upper quadrant likely representing a gallstone within the neck of the gallbladder.'	*Cholecystitis*
'His tip of the appendix is inflamed.'	*Appendicitis*
'The respirations were unlabored and there were no use of accessory muscles.'	*Shortness of breath (NEG)*
'She was walking outside on her driveway and suddenly fell unconcious, with no prodrome, or symptoms preceding the event.'	*Syncope*
'This is important to prevent shortness of breath and lower extremity swelling from fluid accumulation.'	*Edema*

Table 3.1: Example sentences extracted from clinical narratives with implicit entities

[Aronson 2006] [Friedman et al. 1994] [Savova et al. 2010] [Friedman et al. 2004] [Fu and Ananiadou 2014] [Pradhan et al. 2015] would not identify the presence of these entities.

Linking implicitly mentioned entities in clinical narratives is a challenging task. Besides the fact that they lack the proper name of the entity, they can be embedded with negations. For example, the semantics of the sentence *'The patients' respiration become unlabored and there were no use of accessory muscles'* implies that the patient does not have *shortness of breath*. Identifying negations in sentences is referred to as negation detection and it has traditionally been considered as a separate task from the entity linking. This separation is feasible with explicit mentions of the entities since the negation indicating terms are always separable from the phrase indicating entity mention whereas in implicit mentions it is often embedded in the phrases with the entity mentions. For instance, consider two phrases *'patient denies shortness of breath'* and *'patient is breathing comfortably'*. The first phrase has a negated explicit entity mention. The phrase indicating entity ('shortness of breath') and the term indicating negation ('denies') can be identified separately. The second phrase has an implicit negated mention of *shortness of breath*. This phrase uses the term 'comfortably' to indicate the negated mention. The phrase indicating entity and the term indicating negation cannot be identified separately. Hence, this dissertation considers negation detection a sub-task of implicit entity linking. Negation detection in clinical documents is crucial as they provide valuable insights into the patients' health status.

As demonstrated with the above examples, the implicit entity mentions in clinical documents use physiological observations of the clinical conditions to indicate the presence of clinical entities. Hence, one should know the physiological observations related to the entities to identify their implicit mentions. The solution developed in this dissertation uses the knowledge embedded in entity definitions obtained for each entity from the Unified Medical Language System (UMLS) [Bodenreider 2004] to model the entities with their physiological observations. The developed solution: a) acquires relevant entity definitions from UMLS, b) models the entities using their definitions, c) identifies the sentences in input text that may contain implicit entity mentions, and d) performs implicit entity linking by calculating the semantic similarity between entity models and identified sentences.

The solution is evaluated on a ground truth dataset annotated with the help of medical students and shows that the developed approach outperforms the most relevant unsupervised baseline solutions and improves the results of a supervised baseline.

3.2 Related Work

While this dissertation addresses the implicit entity linking (IEL) problem in clinical narratives for the first time, there exist a large body of related research both to the problem and to the solution developed, including Named Entity Recognition (NER), Entity Linking (EL), Paraphrase Recognition, and Textual Entailment Recognition.

Much like IEL, both NER and EL have the objective of binding a natural language expression to a semantic identifier. However, NER and EL solutions expect the proper names (explicit mention) of entities or noun phrases [Collins and Singer 1999] [Bunescu and Pasca 2006]. The solutions developed for NER leverage regularities in morphological and syntactical features that are unlikely to hold in the case of IEL. The most successful NER approaches use word-level features (such as capitalization, prefixes/suffixes, and punctuation), list lookup features (such as gazetteers, lexicons, or dictionaries), as well as corpus-level features (such as multiple occurrences, syntax, and frequency) [Nadeau and Sekine 2007]. These features are common to the explicit entity mentions but they are not exhibited by the phrases that manifest implicit entity mentions in text. The absence of such features challenges the current NER solutions to identify implicit entities. This shortcoming of the NER solutions negatively impacts EL solutions as they typically depend on the output of NER tools [Hachey et al. 2013] to assign a unique identity to the entities. Moreover, state-of-the-art EL approaches include a 'candidate mapping' step that uses entity names to narrow down the space of possible entity identifiers. For example, an EL tool identifies the possible entities that may be signified by the term 'Chicago' in a text by mapping the term 'Chicago' with all known entity names (Chicago city, Chicago movie, Chicago Bulls, etc). This is not possible in the case of IEL due to the absence of the entity names in the text.

Given the inadequacy of syntactic features to address the implicit entity linking problem, this dissertation developed a solution that considers the implied meaning of the text phrases and matches it with the dictionary definitions of the entities. Hence, the tasks of paraphrase recognition [Barzilay and Elhadad 2003] [Dolan et al. 2004] and textual entailment recognition [Giampiccolo et al. 2007] are related to developed solution. However, these tasks are fundamentally different in two aspects: 1) both paraphrase recognition and textual entailment recognition are defined at the sentence level, whereas text phrases considered for IEL can exist as a sentence fragment or span across multiple sentences, and 2) the objective of IEL is to find whether a given text phrase has a mention of an *entity* – as opposed to determining whether two sentences are similar or entail one another. However, the developed solution to link implicit entities benefits from the lessons learned from both tasks.

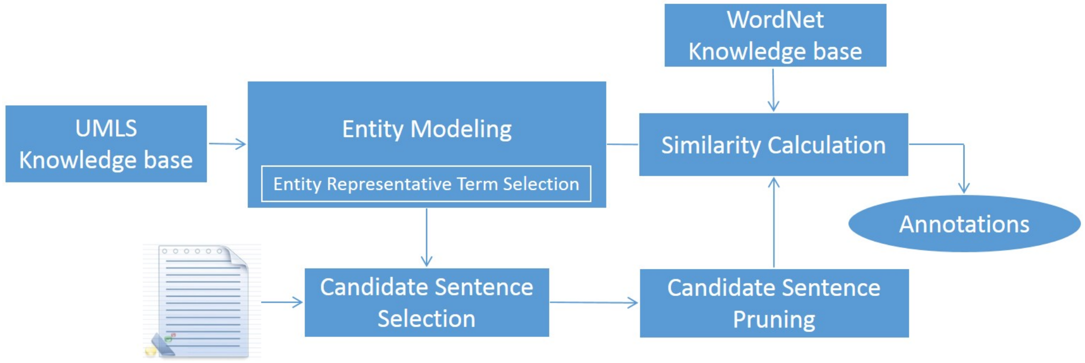

WordNet Knowledge base
UMLS Knowledge base
Entity Modeling
Entity Representative Term Selection
Similarity Calculation
Annotations
Candidate Sentence Selection
Candidate Sentence Pruning

Figure 3.1: Components of the proposed solution

3.3 Implicit Entity Linking in Clinical Narratives

Implicit entity linking in clinical documents is formulated as a classification task, i.e., given an input text, classify it into one of the three categories: TP_e if the text has an implicit mention of entity e, or $Tneg_e$ if the text has a negated implicit mention of entity e, or TN_e if the entity e is not mentioned in the text. As mentioned, the phrases with implicit entity mentions can span multiple sentences. However, this dissertation focuses only on implicit mentions that exist within a single sentence. Figure 3.1 shows the components of the implemented solution, which are discussed below in detail with respect to the framework components described in Section 1.6.

3.3.1 Knowledge Acquisition

As demonstrated with the examples above, the relevant knowledge should consist of physiological observations of clinical entities. The definitions present in UMLS define the clinical entities using their physiological observations. UMLS is a set of files and software that brings together many health and biomedical vocabularies and standards to enable interoperability between computer systems.[1] UMLS consists of three tools: 1) Metathesaurus which consists of terms and codes from many different vocabularies, 2) Semantic Network which consists of broad categories and their relationships, and 3) SPECIALIST Lexicon and Lexical Tools for

[1] https://www.nlm.nih.gov/research/umls/quickstart.html

natural language processing. This dissertation uses the clinical terms and codes defined in the metathesaurus to acquire the required knowledge.

The metathesaurus is a set of relational tables that contain information on entities, including the identity of the entities, definitions of the entities, the semantic types of the entities, and the hierarchical and non-hierarchical relationships between entities. The relational table MRDEF in the UMLS Metathesaurus contains one or more definitions for each clinical entity. These definitions are given as text snippets. Following are the definitions given for the clinical condition *shortness of breath* in MRDEF:

- Difficult or labored breathing.
- Labored or difficult breathing associated with a variety of disorders, indicating inadequate ventilation or low blood oxygen or a subjective experience of breathing discomfort.
- Difficult, painful breathing or shortness of breath.
- A disorder characterized by an uncomfortable sensation of difficulty in breathing.
- An uncomfortable sensation of difficult breathing. It may be present as an acute or chronic sign of an underlying respiratory or heart disorder.

As mentioned, the UMLS is a collection of health and biomedical vocabularies. Each vocabulary has definitions for entities. This has resulted in multiple definitions for some entities in the UMLS. The above definitions for the entity *shortness of breath* complement each other by providing multiple descriptions using a diverse vocabulary. Such definitions provide ideal input to model the entities for the purpose of implicit entity linking as implicit entities manifest in text through different levels of description via diverse vocabulary. Hence, the knowledge acquisition phase of the developed solution retrieves the definitions for the relevant entities from the MRDEF relational table of the UMLS Metathesaurus.

3.3.2 Knowledge Modelling

The knowledge modelling step involve modelling entities of interest. The idea of the entity model is to capture the relevant knowledge about the entity and represent it in a way that can be leveraged to link implicit mentions of entities. The relevant knowledge is present in entity definitions; however, definitions are lengthy and contain supplementary information. Hence, it is necessary to identify and isolate portions of the definitions that characterize the entity. In order to facilitate these tasks, the algorithm introduces the concept of an *entity representative term* associated with each entity and proposes an automatic way to select these terms from entity definitions.

3.3.2.1 Entity Representative Term Selection

Entity representative term (ERT) selection finds a term with high *representative power* for an entity and plays an important role in defining it.

The *representative power* of the term t for entity e is defined based on two properties: its dominance among the definitions of entity e, and its ability to discriminate the entity e from other entities. Consider the entity *appendicitis* as an example. One of its definitions is *'acute inflammation of appendix'*. Intuitively, both terms *inflammation* and *appendix* are clinically relevant terms for explaining the entity *appendicitis*. However, the term *appendix* is more discriminative of an implicit mention of *appendicitis* than the term *inflammation* because the term *inflammation* can be used to describe many other entities. In other words, the term 'appendix' is more specific to *appendicitis* than the term 'inflammation'. Also, none of the definitions of *appendicitis* omit the term *appendix*; therefore, *appendix* is the dominant term, and consequently it has the most representative power for the entity *appendicitis*.

This intuition is captured by a metric inspired by the TF-IDF measure and is formalized in eq. (3.1). The IDF (inverse document frequency) value measures the specificity of a term in the definitions. The TF (term frequency) captures the dominance of a term. Hence the representative power of a term t for entity e (r_t) is defined as

$$R(t, e) = freq(t, \mathcal{Q}_e) * \log \frac{|E|}{|E_t|} \tag{3.1}$$

where $\mathcal{Q}_e$ is the set of definitions of entity e, E is the set of all entities, $freq(t, \mathcal{Q}_e)$ is the frequency of term t in set $\mathcal{Q}_e$, $|E|$ is the size of the set E ($3,962$ in the corpus used for experiments), and $|E_t|$ is the number of entities defined using term t.

The entity representative term is defined using the representative power of the terms in entity definitions.

Definition 3.3.1 (Entity Representative Term)**.** Let $\mathcal{L}_e = \{t_1, t_2, ..., t_n\}$ be the set of terms in the definitions of an entity e. We select term t_m as the entity representative term of the entity e if its representative power is maximum, i.e., $R(t_m, e) \geq R(t_i, e)$ for all i where $1 \leq i \leq n$.

The selected entity representative term (ERT) for each entity is expanded by adding its synonyms obtained from WordNet. For example, the ERT selected for the entity *shortness of breath* is 'breathing', and this can be expanded by adding the terms 'respiration' and 'ventilation' in WordNet.

3.3.2.2 Modeling Entities with Definitions

Each entity has multiple definitions. Each definition is used to create an *entity indicator*. An entity indicator consists of terms from a definition that characterize the entity. Consider the definition *'A disorder characterized by an uncomfortable sensation of difficulty in breathing'* for the entity *shortness of breath*, for which the selected ERT is *'breathing'*. The terms *uncomfortable*, *sensation*, *difficulty*, and *breathing* collectively describe the entity. Other terms in the definition are supplementary terms and are not required to describe the physiological observations of the entity *shortness of breath*, hence should be ignored when creating the entity model. The entity modelling algorithm exploits the neighborhood of the ERT in the definition to create the entity indicators and automatically selects the nouns, verbs, adjectives, and adverbs in the definition within a given window size to the left and to the right of the ERT. The selected terms and their representative power characterizes the entity indicator. The experiments conducted in this dissertation use a window size of four.

A collection of entity indicators created using definitions of the entity constitutes the *entity model*. In other words, an entity model consists of multiple entity indicators that capture diverse and orthogonal ways

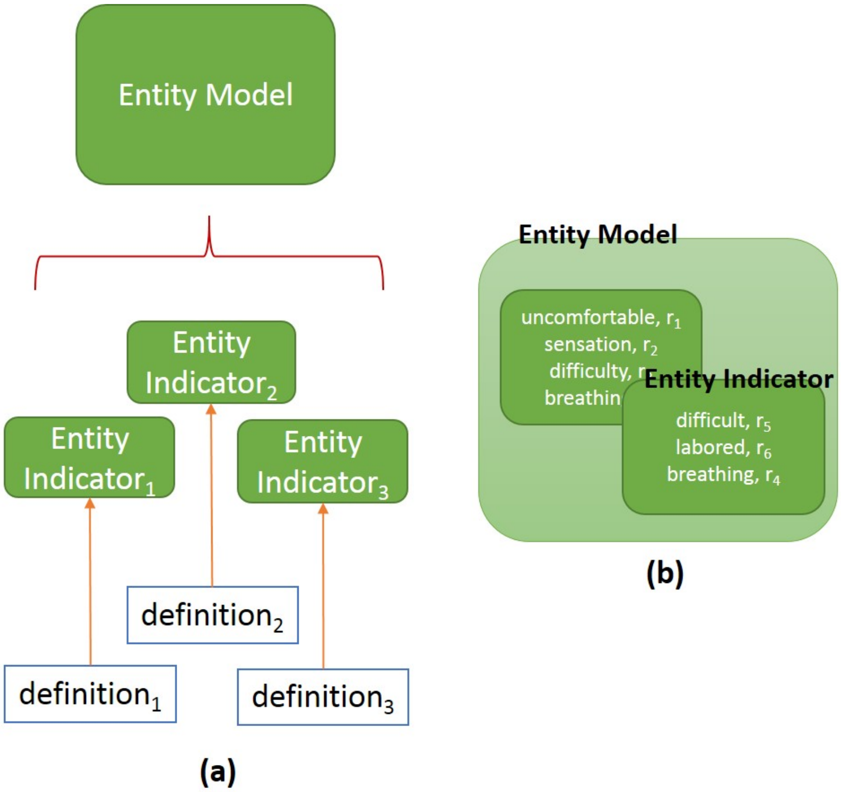


Figure 3.2: (a) Entity model creation, (b) Example entity model

of expressing an entity in the text. Figure 3.2(a) shows the structure and the components of the entity model and Figure 3.2(b) shows an example entity model created for *shortness of breath*. The r_i represents the representative power the terms.

3.3.3 Detecting Sentences with Implicit Entities

The previous steps built the infrastructure that is needed to perform implicit entity linking. The next step is to identify the candidate sentences with potential implicit entity mentions. The ERT selected for each entity helps to identify the candidate sentences. The sentences that contain the ERT of an entity but not the name of the entity in an input text are identified as *candidate sentences* that potentially contain implicit mention of the corresponding entity. A sentence may contain multiple ERTs and consequently become a candidate sentence for multiple entities.

As in the definitions of the entities, candidate sentences can contain supplementary information and also irrelevant information. Hence, the candidate sentences are pruned to focus on the relevant portion of the sentence. The pruning steps are the same as the pruning steps for the entity definitions (i.e., the nouns, verbs, adjectives, and adverbs are selected within a predefined window size to the left and right of the ERT).

3.3.4 Implicit Information Extraction - Linking Implicit Entities in Clinical Narratives

As the last step, the solution calculates the semantic similarity between the entity model and the pruned candidate sentence to classify the sentence as TP_e, $Tneg_e$, or TN_e. Sentences with implicit entity mentions often use adjectives and adverbs to describe the entity and they may indicate the absence of the entities using antonyms or explicit negation. These two characteristics pose challenges to the applicability of existing text similarity algorithms such as MCS [Mihalcea et al. 2006] and matrixJcn [Fernando and Stevenson 2008] which are proven to perform well among the unsupervised algorithms in paraphrase identification task [ACLWiki 2014].

The existing text similarity algorithms largely benefit from the WordNet similarity measures. Most of these measures use the semantics of the hierarchical arrangement of the terms in WordNet. Unfortunately, adjectives and adverbs are not arranged in a hierarchy, and terms with different part-of-speech (POS) tags cannot be mapped to the same hierarchy. Hence, they are limited in calculating the similarity between terms of these categories. This limitation negatively affects the performance of IEL as the entity models and pruned sentences often contain terms from these categories. Consider the following examples:

1. *Her breathing is still uncomfortable*$_{adjective}$.
2. *She is breathing comfortably*$_{adverb}$ *in room air.*
3. *His tip of the appendix was inflamed*$_{verb}$.

The first two examples use an adjective and an adverb to mention the entity *shortness of breath* implicitly. The third example uses a verb to mention the entity *appendicitis* implicitly instead of the noun *inflammation* that is used in its definition.

This dissertation has developed a text similarity measure that overcomes these challenges and weighs the contributions of the words in the entity model in relation to the similarity value based on their representative power.

Handling adjectives, adverbs, and words with different POS tags: To get the best out of all WordNet similarity measures, the solution exploited the relationships between different forms of the terms in WordNet to find the noun form of the terms in the entity models and pruned sentences before calculating the similarity. The adjectives for adverbs were found using relationship 'pertainym', and nouns for adjectives or verbs were found using the relationship 'derivationally related form' in WordNet.

Handling negations: Negations are of two types: 1) negations mentioned explicitly with terms like no, not, and deny, and 2) negations indicated with antonyms (e.g., 2nd example in the above list). The NegEx algorithm [Chapman et al. 2001] is used to address the first type of negation. The antonym relationships in WordNet's linguistic knowledge base are used to address the second type of negation.

The similarity between an entity model and the pruned candidate sentence is calculated by computing the similarities of their terms. The term similarity is computed by forming an ensemble using the standard WordNet similarity measures — namely, WUP [Wu and Palmer 1994], LCH [Leacock and Chodorow 1998], Resnik [Resnik 1995], LIN [Lin 1998], and JCN [Jiang and Conrath 1997], as well as a predict vector-based measure, Word2vec, [Mikolov et al. 2013] and a morphology-based similarity metric, Levenshtein[2], as:

$$sim(t_1, t_2) = max_{m \in M}(sim_m(t_1, t_2)) \tag{3.2}$$

where t_1 and t_2 are input terms and M is the set of above mentioned similarity measures. This ensemble-based similarity measure exploits orthogonal ways of comparing terms: semantic, statistical, and syntactic. An ensemble-based approach is preferable over picking one of them exclusively since they are complementary in nature; that is, each outperforms the other two in certain scenarios.

The similarity values calculated by WordNet similarity measures in $sim_m(t_1, t_2)$ are normalized to range between 0 and 1.

[2] http://en.wikipedia.org/wiki/Levenshtein_distance

The similarity of a pruned candidate sentence to the entity model is calculated by calculating its similarity to each entity indicator in the entity model, and picking the maximum value as the final similarity value between the entity model and the candidate sentence. The similarity between entity indicator ei and pruned sentence s, $sim(ei, s)$, is calculated by summing the similarities calculated for each term t_{ei} in the entity indicator weighted by its representative power as defined in eq. (3.1). If t_{ei} is an antonym for any term in s (t_s), it contributes negatively to the overall similarity value, else it contributes in linear portion to the maximum similarity value between t_{ei} and some t_s (eqs. (3.4) and (3.5)). The overall similarity value is normalized based on the total representative power of all the terms in ei (eq. (3.1)) and ranges between -1 and +1.

Note that this formulation weighs the contribution of each term according to its importance in defining the entity. The higher the similarity with a term that has a higher representative power leads to a higher overall similarity value, while the lower similarity with such terms leads to a lower total similarity value. The special treatment of antonyms takes care of the negated mentions of an entity.

$$sim(ei, s) = \frac{\sum_{t_{ei} \in ei} f(t_{ei}, s) * R(t_{ei}, e)}{\sum_{t_{ei} \in e} R(t_{ei}, e)} \tag{3.3}$$

$$f(t_{ei}, s) = \begin{cases} -1 & \alpha(t_{ei}, s) == 0 \\ \max_{t_s \in s} \mathrm{sim}(t_{ei}, t_s) & \text{otherwise} \end{cases} \tag{3.4}$$

$$\alpha(t_{ei}, s) = \prod_{t_s \in s} \begin{cases} 0 & \text{if } t_{ei} \text{ is an antonym of } t_s \\ 1 & \text{otherwise} \end{cases} \tag{3.5}$$

In order to capture the explicit negations, the NegEx algorithm is ran on the sentence and added -1 to the similarity if NegEx detects a negation. Finally, the sentences are classified based on a configurable threshold value selected between -1 and +1. This solution is referred to as an implicit entity linking solution (IEL solution) from here onwards.

3.4 Evaluation

There is no gold standard dataset readily available to evaluate this task. Hence, a new gold standard dataset is created by annotating a sample dataset from the corpus created for SemEval-2014 task 7 [Pradhan et al. 2014]. This dataset is used to evaluate the performance of IEL solution in classifying entities annotated with *TP* and *Tneg* mentions.

3.4.1 Gold Standard Dataset

The SemEval-2014 task 7 corpus consists of 24,147 de-identified clinical notes. This corpus is used to create a gold standard with the help of three domain experts to evaluate implicit entity linking. The gold standard consists of 857 sentences selected for eight entities. The creation of the gold standard is described below in detail.

The first task in creating the gold standard is to select entities of interest. Such entities are selected based on the frequency of their appearance in the corpus. In order to find frequent entities, the corpus is annotated with cTAKES [Savova et al. 2010] and the entities are ranked based on their frequency. cTAKES is a software solution developed to extract information from electronic medical records. It annotates clinical entities in electronic medical records by using the Unified Medical Language System (UMLS) as the reference knowledge base. Its annotations include disorders, symptoms, procedures, anatomical structures, and medications. In addition to the entity annotations, cTAKES extracts the metadata associated with these entities in clinical documents. For example, it is able to identify whether the disorder is negated in the document, whether the disorder is related to the current diagnosis or the patient's history, whether the disorder is related to patient's clinical history or family clinical history, etc.

The entity annotations produced by cTAKES on the sample dataset were ranked based on the frequency. The domain experts then selected a subset from the frequent entities that they judged to be frequently mentioned implicitly in clinical documents according to their experience. For example, the frequent entity *shortness of breath* was selected but not *chest pain* since the former is mentioned implicitly often but not the latter.

Entity	**TP**	**Tneg**	**TN**
Shortness of breath	93	94	29
Edema	115	35	81
Syncope	96	92	24
Cholecystitis	78	36	4
Gastrointestinal gas	18	14	5
Colitis	12	11	0
Cellulitis	8	2	0
Fasciitis	7	3	0

Table 3.2: Candidate Sentence Statistics

This study helped to choose four entities as the primary focus of the evaluation and will refer to these as *primary entities* from here on (the first four entities in Table 3.2). To test the generalizability of IEL solution, as well as to evaluate its robustness when lacking training data, another four entities (the last four entities in Table 3.2) were selected for the evaluation.

The next task is to create a dataset that consists of sentences with implicit mentions of these selected entities. This is accomplished by selecting a random sample of candidate sentences which contain the entity representative term selected for each entity of interest. The selected random sample is winnowed down further by manually selecting a subset that exhibits syntactic diversity and is composed of diverse vocabulary as too many sentences with similar syntax and similar vocabulary would bias the results obtained. Ultimately, our corpus consisted of 120-200 sentences for each primary entity and an additional 80 sentences selected for the other four entities.

Each selected sentence was annotated as TP_e (contains a mention of entity e), $Tneg_e$ (contains a negated mention of entity e), or TN_e (does not contain a mention of entity e). Each sentence was annotated by two domain experts, and we used the third domain expert to break the ties. The Cohens' Kappa value for the annotation agreement was 0.58. While the annotators have good agreement on annotating sentences in category *TP*, they agreed less on the categories *Tneg* and *TN*. The latter categories are indeed difficult to distinguish. For example, annotators often argue whether *'patient breathing at a rate of 15-20'* means the negation of the entity *shortness of breath* (because that is a normal breathing pattern) or if it just lacks a mention of the entity. The final annotation label for a sentence is decided based on majority voting. Table 3.2 shows the statistics of the annotated candidate sentences.

3.4.2 Implicit Entity Linking Performance

There are no baseline algorithms that can be directly applied to link implicit entities such that it would yield a fair comparison with the IEL solution. However, there are a few related algorithms that potentially can be applied to this task. Therefore, two strong algorithms were selected from the closest related work as baseline solutions to the problem.

The first baseline is the well-known text similarity algorithm MCS [Mihalcea et al. 2006]. MCS is one of the best performing unsupervised algorithms in paraphrase recognition task [ACLWiki 2014]. The MCS algorithm takes two sentences S_1 and S_2 as input and calculates the similarity between each word of S_1 with each word of S_2 using WordNet similarity measures, pointwise mutual information (PMI), and latent semantic analysis. It considers the maximum similarity value given by these measures for each pair of words for the final similarity calculation. It weighs and normalizes the similarity values based on the inverse document frequency (IDF) value of the word generated from a large corpora. The MCS similarity measure is formalized in Equation (3.6).

$$sim(S_1, S_2) = \frac{1}{2}(\frac{\sum_{w \in S_1}(maxsim(w, S_2) * idf(w))}{\sum_{w \in S_1} idf(w)} + \frac{\sum_{w \in \{S_2\}}(maxsim(w, S_1) * idf(w))}{\sum_{w \in S_2} idf(w)}) \quad (3.6)$$

Both MCS and IEL solution classify the candidate sentences based on threshold values selected experimentally.

Support Vector Machine [Cortes and Vapnik 1995] is selected as a supervised baseline. SVM is one of the state-of-the-art learning algorithms shown to perform remarkably well in number of classification tasks.

SVM is trained for each primary entity considering unigrams, bigrams, and trigrams as the features. This preference is motivated by the fact that SVM trained on ngrams performed well on text classification tasks [Pang et al. 2002] [Zhang and Lee 2003]. The SVMs trained with bigrams consistently produced the best results for the 4-fold cross validation in the implicit entity linking task. Therefore, the testing phase used SVMs trained with the bigrams as the supervised baseline.

Preparation of training and testing datasets: The training and testing datasets were created by splitting the dataset annotated for each primary entity as 70% (training) and 30% (testing). The training datasets were used to train the SVM models for each primary entity and to select the threshold values for both MCS algorithm and IEL solution.

The classification performance of each algorithm is studied in the *TP* and *Tneg* categories using precision, recall, and F-measure.

The precision (*PP*) and recall (*PR*) for category *TP* at threshold t are defined as:

$$PP_t = \frac{S_{TP}\ with\ sim \geq t}{all\ sentences\ with\ sim \geq t} \tag{3.7}$$

$$PR_t = \frac{S_{TP}\ with\ sim \geq t}{S_{TP}} \tag{3.8}$$

Similarly, *NP* and *NR* for *Tneg* are defined as:

$$NP_t = \frac{S_{Tneg}\ with\ sim < t}{all\ sentences\ with\ sim < t} \tag{3.9}$$

$$NR_t = \frac{S_{Tneg}\ with\ sim < t}{S_{Tneg}} \tag{3.10}$$

where S_{TP} and S_{Tneg} denote the sentences annotated with *TP* and *Tneg* respectively by domain experts and sim is the calculated similarity value for the pruned sentence.

Selecting threshold value: The threshold values for both MCS and IEL solution are selected based on their classification performance in the training dataset. The MCS algorithm produced the best F1 score for the *TP* category with a threshold value of 0.5, and for the *Tneg* category with a threshold value of 0.9, while IEL solution produced the best F1 for the *TP* category with 0.4 and for the *Tneg* category with 0.3. We examined threshold values that produce the best F1 scores by the two algorithms by starting with 10% of the training data and gradually increasing the size of the training data. The threshold values with the best F1 scores were stabilized after adding 30% of the training data.

3.4.3 Classification Performance

The first experiment evaluates the performance of the classification task of IEL solution, MCS, and SVM. Table 3.3 shows the results obtained for two baseline methods and the IEL solution. IEL solution outperforms the MCS algorithm, but the SVM was able to leverage supervision to outperform IEL solution in the *TP* category in terms of F-measure ($PF1$ on Table 3.3). For example, the sentence *'he was placed on mechanical ventilation shortly after presentation'* is annotated as *TP* in the gold standard for the entity *shortness of breath* since *'mechanical ventilation'* indicates the presence of *shortness of breath*. This annotation requires domain knowledge that was not present in the entity definitions that we used to build the entity models. However, with enough examples, the SVM was able to learn the importance of the bigram *'mechanical ventilation'* for the entity *shortness of breath* and classify it as *TP*.

Method	**PP**	**PR**	**PF1**	**NP**	**NR**	**NF1**
IEL Solution	0.66	0.87	0.75	**0.73**	0.73	**0.73**
MCS	0.50	**0.93**	0.65	0.31	**0.76**	0.44
SVM	**0.73**	0.82	**0.77**	0.66	0.67	0.67

Table 3.3: precision, recall, and F1 values for each algorithm (PF1 and NF1 indicate F1 scores for the *TP* and *Tneg* categories respectively).

However, IEL solution outperforms the SVM in the *Tneg* category ($NF1$ on Table 3.3). This is due to the explicit treatment of the negated mentions by IEL solution to capture different variations of the negations.

The MCS algorithm underperformed in both categories. It is observed that this was mostly due to its limitations described in Section 3.3.4; namely, 1) an inability to handle adjectives, adverbs, and words with different part-of-speech tags, and 2) the absence of a mechanism to address the negations. The overall classification accuracy—the accuracy of classifying both *TP* and *Tneg* instances—of IEL solution, MCS, and SVM were 0.7, 0.4, and 0.7 respectively.

Method	**PP**	**PR**	**PF1**	**NP**	**NR**	**NF1**
SVM	0.73	0.82	0.77	0.66	0.67	0.67
SVM+MCS	0.73	0.82	0.77	0.66	0.66	0.66
SVM+IEL Solution	**0.77**	**0.85**	**0.81**	**0.72**	**0.75**	**0.73**

Table 3.4: Comparison of SVM results incorporating similarity values calculated by our algorithm and MCS as a feature.

The second experiment evaluates the impact of including the similarity scores calculated by MCS and IEL solution for each candidate sentence as a feature to the best performing SVM model. Table 3.4 shows the results of this experiment. Table 3.4 shows that the inclusion of MCS scores as a feature did not help to improve the SVM results. In fact, it negatively affected the results for the *Tneg* category. Since the MCS showed a low precision for the *Tneg* category in the previous experiment (Table 3.3), it is potentially introducing too much noise that the SVM is not able to linearly separate. However, the similarity value calculated by IEL solution improves the SVM classifiers. It increased the precision and recall values for both the *TP* and *Tneg* categories. This shows that the similarity value calculated by IEL solution can be used as an effective feature for a learning algorithm that is designed to solve the IEL problem.

These evaluations were conducted with the sentences selected for primary entities. The sentences collected for other entities were used to evaluate the generalizability of the IEL solution. There were 80 sentences selected for non-primary entities (the last four entities in Table 3.2). IEL solution produced the following results for these entities when the classification was performed with the threshold values selected using the training dataset created for the primary entities.

$$PP = 0.72, PR = 0.77, PF1 = 0.74$$

$$NP = 0.78, NR = 0.83, NF1 = 0.80$$

These results demonstrate the generalizability of the threshold value selected using primary entities to the sentences of other entities.

Although negation detection with NegEx is not a contribution of IEL solution, it enables the application of NegEx to IEL. This is not possible for MCS. NegEx requires two inputs: 1) the sentence, and 2) the term being considered for possible negation. MCS does not detect the key term in the sentence, hence it is not possible to apply NegEx with MCS. However, IEL solution starts with identifying the entity representative term which is considered for negation.

3.4.4 A Study on Volume of Training Data Requirement

Although the supervised learning technique outperformed IEL solution in the *TP* category, training the SVM model for each entity separately is required. This is a time and resource consuming task since it requires labeled data for each entity. The above experiments showed that the similarity value calculated by IEL solution can enhance the result of the SVM classification. This observation motivates the hypothesis that the similarity value calculated can also be used to reduce the amount of labeled data needed to achieve results similar to IEL solution with supervision. Hence, this evaluation studied the variation of the volume of training data needed by different configurations of the SVM to obtain the results obtained by IEL solution. It considered three configurations of the SVM and analyzed their behavior with IEL solution with different training dataset sizes. The three configurations were 1) SVM trained on bigrams, 2) SVM trained on bigram with MCS similarity value as a feature, and 3) SVM trained on bigram with IEL similarity value as a feature. The F1 value of the classification results was obtained by varying the size of the training data in these three configurations. Figure 3.3 shows the F1 values obtained by gradually increasing the size of the training dataset[3], while testing on the same test set.

The F1 value of the IEL solution remains constant after 50% of the training data has been provided since it has already decided upon the threshold values. Figure 3.3 shows that the SVM trained with bigrams needs 76% of the training set to achieve the F1 value achieved by IEL solution in the *TP* category, and it

[3] The graphs were drawn considering training dataset size >50% for clarity.

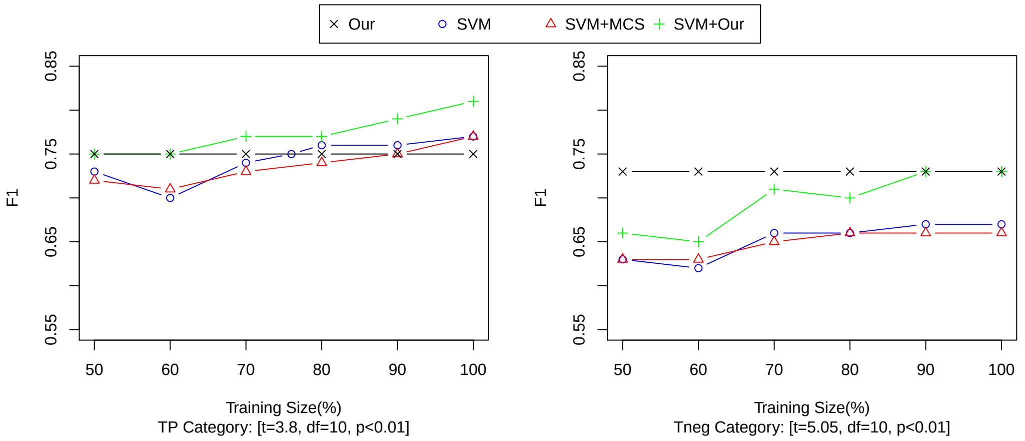


Figure 3.3: The variation of the F1 value in the *TP* (left) and *Tneg* (right) categories with varying sizes of training datasets

does not achieve the F1 achieved by IEL solution in the *Tneg* category (note: the crossing points of the line marked with ‘X’ and line marked with circles). Figure 3.3 also shows that the similarity score calculated by IEL solution complements the SVM at each data point. After adding the similarity score calculated by IEL solution to SVM as a feature, it crossed the F1 of the IEL solution with just 50% of the training data in the *TP* category and with 90% of the training data in the *Tneg* category (note the crossing points of the line marked with ‘X’ and line marked with ‘+’). The paired T-test values calculated for SVM and SVM+IEL solution configurations show that this is not a random behavior (t - T-test value, df - degree of freedom, p - probability value). Also, the SVM+IEL solution configuration achieved the best F1 value for SVM with just 70% of the training data in *TP* category and with just 50% of the training data in the *Tneg* category (compare the highest point in the line marked with the circle with the data points of the line marked with ‘+’). This shows that the similarity value calculated by the IEL solution can be used as an effective feature to reduce manual labeling effort and to improve the supervised learning algorithms used in solving the IER problem.

3.4.5 Variation of the Classification Performance with Annotation Confidence

Each annotation generated by the domain expert is given a confidence value. This value reflects the confidence of the domain expert in the annotation decision. This confidence value is a function of the clarity of the sentence and the completeness of the information in the sentence, and it is ranges between 1 and 5, with 1 being the lowest confidence and 5 being the highest confidence. Hence, it is interesting to analyze the variation of the classification results with the confidence.

Figure 3.4 shows the F1 value of the classification results obtained by IEL solution for positive sentences and negative sentences separately in different confidence ranges. The x-axis signifies the range of the confidence value, i.e., each data point signifies the F1 value of the sentences annotated with confidence greater than its x-value. For example, the top right triangle indicates the F1 value of classifying sentences annotated with *Tneg* and confidence > 4. As expected, the F1 value increases as the annotation confidence increases. This increment is much more noticeable with the sentences annotated as *Tneg*.

3.5 Limitations

The candidate sentence selection based on the entity representative term can be seen as a limitation of our approach since it does not select sentences with implicit entity mentions that do not use the selected ERT. It turns out that this observation minimally affects candidate sentence selection. The domain experts were asked to come up with the sentences that contain implicit mentions of the entity *shortness of breath* without using its ERT *'breathing'* or its synonyms (*'respiration'* and *'ventilation'*). According to them, the sentences *'the patient had **low oxygen saturation**'*, *'the patient was **gasping for air**'*, and *'patient was **air hunger**'* are such sentences (the emphasis indicates the phrases that imply *shortness of breath*). However, it is found that there are only 113 occurrences of these phrases as opposed to 8990 occurrences of its ERTs in our corpus. As another example, consider the sentence *'The patient fell and was **initially non responsive**'* which has an implicit mention of entity *syncope*. The selected ERT for syncope is *'consciousness'* and it is not present in the above sentence. However, we found only 71 such sentences in the corpus of 24,147 clinical documents.

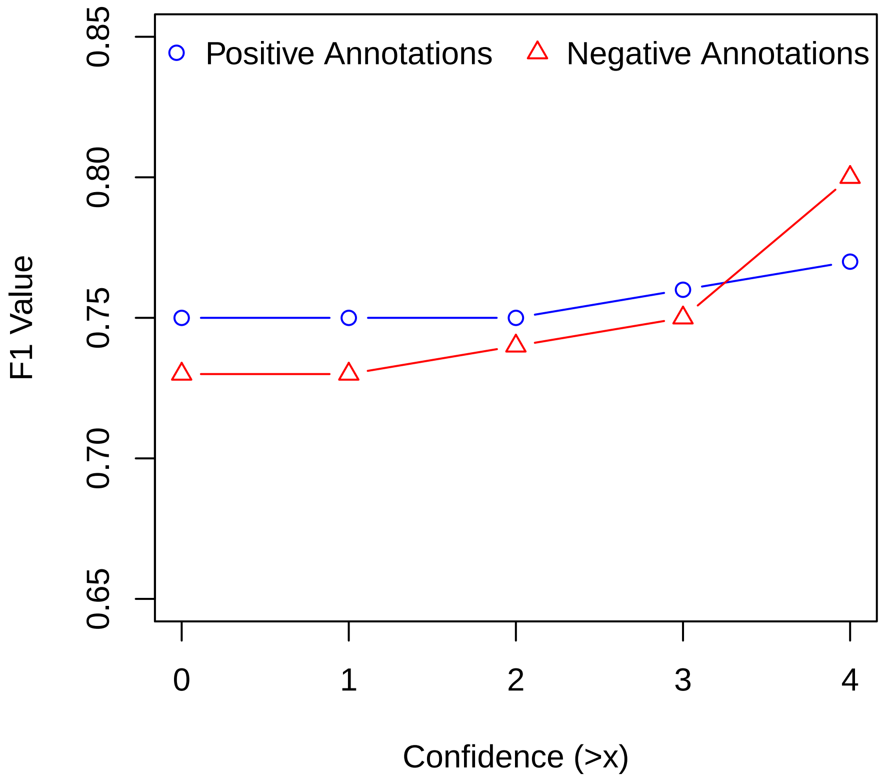


Figure 3.4: Variation of classification performance with annotation confidence.

This study suggests that our selection of ERT only marginally affects the recall of the IER performance in a negative manner.

3.6 Conclusion

Implicit entity mentions are a common occurrence in clinical documents as they are often typed during a patient visit in a way that is natural in spoken language and meant for consumption by professionals with similar backgrounds. Current state-of-the-art solutions do not recognize the implicitly mentioned entities in clinical text. Failure to link implicitly mentioned entities in clinical text can negatively impact the downstream applications.

This chapter introduces the problem of *implicit entity linking* in clinical documents. It studied the characteristics of implicit entity manifestations of clinical entities in text. It used the knowledge available in the UMLS vocabulary to model the entities and developed a semantic similarity measure that is capable of comparing the entity models and candidate sentences to decide upon the presence or absence of implicit entity mentions. The developed solution outperformed the current applicable unsupervised solution to link implicit entities in clinical text. The developed similarity measure can also be used to improve the results of supervised solutions and reduce the need for labeled data for training the models.

4

Implicit Entity Linking in Tweets

4.1 Overview

Tweeting is a very popular mode of communication. Data show that 350,000 tweets are generated per minute – 500 million per day.[1] These tweets have become a valuable source of information for various applications, including trend detection, event monitoring, and opinion mining. All these applications need to perform some variation of an information extraction step in order to get the relevant information from a corpus of tweets. The information extraction task on tweets poses unique challenges due to their short, noisy, context-dependent, and dynamic nature [Derczynski et al. 2015].

Identifying and extracting entities from text is a very common information extraction task among all these applications. State-of-the-art entity linking solutions in tweets have mainly focused on explicitly mentioned entities [Chang et al. 2014] [Derczynski et al. 2015] [Guo et al. 2013] [Liu et al. 2013]. However, as shown in Chapter 1, entities are also mentioned implicitly in tweets. Failure to identify the implicit mentions of entities in tweets negatively impacts the applications that use tweets as a data source. This chapter introduces the problem of implicit entity linking in tweets and highlights their characteristics, which are very different from the implicit entity manifestations in clinical narratives.

The set of entities that potentially can appear in clinical narratives can be obtained from a vocabulary,

[1] http://www.internetlivestats.com/twitter-statistics

and this set largely remains same. The implicit entities in clinical narratives manifest with different syntactic and semantic variations of their definitions. Hence, the main challenge is to develop a method to bridge the syntactic and semantic gap between the entity definitions and manifestations of implicit entities in text. However, when it comes to tweets, the set of entities that appear in tweets change over time in response to real-world events. The entities that appear in tweets do not have definitions curated by humans as in the case of clinical entities. Further, the way people express the presence of implicit entities in tweets changes overtime and awareness of the events happening in the real world is needed to decode the messages conveyed in tweets.

This dissertation addresses these challenges in linking implicit entity mentions in tweets. Twitter users often rely on sources of context outside the current tweet, assuming that there is some shared understanding between them and their audience, or shared temporal context in the form of recent events or recently-mentioned entities [Derczynski et al. 2015]. This assumption allows them to constrain the message to 140 characters, yet make it understandable to the audience. The algorithm developed in this chapter models entities by encoding this shared understanding by harnessing the factual and contextual knowledge of entities to complement the context expressed in the tweet text. The knowledge required to build the shared understanding is obtained by querying structured knowledge bases and mining daily communications on Twitter. The knowledge obtained is used to generate entity models and these entity models are used to identify and link the implicit entities in tweets. The entity linking problem is formulated as a ranking problem and the developed solution learns ranking models in a supervised manner from the data.

4.2 Related Work

Entity linking in tweets has recently gained attention in academia and industry alike. The literature on entity linking in tweets can be categorized as 'word-level entity linking' and 'whole-tweet entity linking' [Derczynski et al. 2015]. While the former task is focused on resolving the entity mentions in a tweet, the latter task is focused on deriving the topic of the tweet. The topic may be derived based on the explicit and implicit entities mentioned in tweets. For instance, the tweet *'Texas Town Pushes for Cannabis Legalization to Combat Cartel Traffic'* has an explicit mention of entity *Cannabis Legalization* and an implicit mention of the entity the city of *El Paso*. The topic of the tweet would be 'Cannabis Legalization in El Paso.' Hence, it is worth noting that the work on deriving topics is comparable neither to explicit nor to implicit entity linking since they are extracting the topic of the tweet text rather than actual mentions of entities in the tweet [Derczynski et al. 2015]. This section surveys the literature on both word-level and whole-tweet entity linking and explains why techniques and features used by such solutions may not be applicable to link implicit entities in tweets.

Meij, et al. [Meij et al. 2012] derive the topics of a tweet. It extracts features from the tweet and the Wikipedia pages of entities, and applies machine learning algorithms to derive the topic. [Meij et al. 2012] have focused on deriving topics using explicit entities since the evaluation dataset contained only 16 tweets whose label of the manually annotated topic was not present in the tweet text (i.e., not a string match). Nevertheless, they were found to be either synonyms or related entities to the explicit entities in the tweets and not implicit entity mentions (e.g., New York and Big Apple, Almighty and God, stuffy nose and Rhinitis).

Word-level tweet linking has two main steps: candidate selection and disambiguation. Word-level entity linking has been studied extensively for organized text like Wikipedia and news [Cucerzan 2007] [Dredze et al. 2010] [Hoffart et al. 2011] [Mendes et al. 2011] [Milne and Witten 2008]; however, these approaches have proved to be ineffective on short, noisy, and unorganized text like tweets [Derczynski et al. 2015] [Ferragina and Scaiella 2010]. This discussion focuses on approaches taken to solve this problem in tweets. The first step was to match the word sequences of the tweet to the page titles and the anchor texts in Wikipedia and considering all matching pages and pages redirected by matching anchor texts to be candidates [Chang et al. 2014] [Ferragina and Scaiella 2010] [Guo et al. 2013]. The second step was to optimize the relatedness calculated among the candidate entities [Ferragina and Scaiella 2010] [Liu et al. 2013], based on the threshold defined over measures that calculate the similarity between the entity mention and entity representation [Chang et al. 2014], or by applying structural learning techniques [Guo et al. 2013]. An approach to solve implicit entity linking in tweets has to take a fresh perspective since, by definition, neither anchor text nor page title is present in the tweet. Hence, it cannot execute the candidate selection and disambiguation steps in this fashion.

4.3 Implicit Entity Linking in Tweets

Implicit entity linking in tweets in this dissertation focuses on movie and book mentions in tweets. The entities discussed in tweets change over time, hence the solution developed considers entities of interest at particular time t and focuses on linking these entities. The developed solution consists of four components that mirrors the steps of acquiring knowledge, modeling the knowledge, detecting the tweets with implicit entity mentions, and linking the mentioned entities.

The first step of the entity linking solution is to identify the entities of interest that are relevant at time t. This can be done, for instance, by running an off-the-shelf entity linking solution over a corpus of tweets collected at t and identifying the mentioned entities. The idea is that if an entity is relevant at time t, it will likely be mentioned explicitly by tweets around that time. Even though an automatic entity linking solution may not be perfect, it will identify at least one occurrence of an explicit mention of each entity within the corpus. This is sufficient as the focus is to find the entities that are being discussed at t regardless of their frequency. Once the entities of interest are known, the entity linking solution acquires the knowledge about these entities and generates entity models.

4.3.1 Knowledge Acquisition

Consider two tweets with an implicit mention of the movie *Gravity*: *'New Sandra Bullock astronaut lost in space movie looks absolutely terrifying,'* and *'ISRO sends probe to Mars for less money than it takes Hollywood to send a woman to space.'* The first tweet has a mention of its actress and that, along with other terms, help to resolve its mention to the movie *Gravity*. This kind of **factual knowledge** (e.g., to relate actors and movies) can be extracted from a knowledge base such as DBpedia. The second tweet does not have a mention of any such entity associated with the movie *Gravity*; hence, the factual knowledge is not applicable in identifying its implicit mention. This particular tweet associates the budget of the Mars orbiter mission (the space probe sent by Indian space research organization (ISRO) to Mars)[2] to the budget of the movie *Gravity*. Hence, one should know the association between these two events to decode the implicit mention of the movie *Gravity* in this tweet. Such associations can be established by monitoring the topics discussed in contemporary tweets with explicit mentions of the movie *Gravity* as they often use phrases like 'ISRO', 'woman to space', and 'less money'. This dissertation refers to such knowledge component as **contextual knowledge** as it heavily depends on the topics being associated with entities at time t.

4.3.1.1 Acquiring Factual Knowledge

Factual knowledge of an entity can be acquired from existing knowledge bases and Linked Open Data. This dissertation uses DBpedia [Lehmann et al. 2014] as the knowledge base due to its wide coverage of domains and up-to-date knowledge. DBpedia is a generic knowledge base created by extracting the facts from Wikipedia. It consists of both schema in the form of an ontology and facts about the world. It publishes these facts as RDF (Resource Description Framework) triples using its ontology. The latest statistics for DBpedia show that it contains more than 4.5 million entities and over 3 billion facts.[3]

For a given entity e the knowledge acquisition step retrieves triples where e appears as subject or object in DBpedia. However, for a given entity type not all relationships are important in modeling its entities. For example, a movie has the relationships 'director' and 'starring' as well as 'billed' and 'license.' The former two relationships are more important when describing a movie than the latter two. This intuition is captured by ranking the relationships based on their joint probability value with the given entity type as follows.

$$P(r,T) = \frac{\textit{number of triples of r with instances of T}}{\textit{total number of triples of r}}, \tag{4.1}$$

where T is the entity type (e.g. Movie) and r is the relationship. The instances of a given entity type can be obtained from DBpedia via 'rdf:type' relationship.

The triples of entity e of type T with one of the top m relationships are selected to build the entity

[2] https://en.wikipedia.org/wiki/Mars_Orbiter_Mission

[3] http://wiki.dbpedia.org/about/facts-figures

model of e. This step collects the 'rdfs:label' value of entities connected to e in these triples as the factual knowledge of entity e. In addition to the top *m* relationships, it also considers the value of the 'rdfs:comment' relationship of e to build the entity model. 'rdfs:comment' gives a textual description of an entity that often times complements the knowledge captured by triples.

4.3.1.2 Acquiring Contextual Knowledge

Contextual knowledge can be extracted from contemporary tweets that explicitly mention an entity. The knowledge acquisition step uses the rdfs:label value of the entity in DBpedia along with its type as the keyword to collect the 1,000 most recent tweets for that entity. For example, it will use the phrases 'gravity movie' and 'gravity film' to collect tweets for the movie *Gravity*; this selection of keywords minimizes the tweets with other meanings of the term 'gravity' in the collected tweets.

The next step is to extract knowledge from the collected tweets. 'Knowledge' here refers to the meaningful phrases in tweets that may signify entities, events, etc. In order to execute this step, a vocabulary consisting of meaningful phrases is created and matched with the n-grams extracted from tweets. The vocabulary is created by retrieving the page titles and text phrases that appear as anchor text in Wikipedia. Wikipedia pages represent things that have a presence in the real world such as entities and events, and anchor text in Wikipedia pages always link to another page. Hence, anchor text can be considered as phrases representing things in the real world. This qualifies them to be added to the vocabulary of meaningful phrases. The knowledge extraction step chunks the tweet into n-grams (n=2, 3, 4) and the n-grams that match with the phrases in the vocabulary are extracted as knowledge. However, Twitter users do not always use complete phrases; consider the reference to actress *Sandra Bullock* in the tweet: *'It's hard for me to imagine movie stars as astronauts, but the movie looks great! and who doesn't like **Sandra**.'* Therefore, this step includes unigrams, excluding stop words, in the extracted knowledge. This knowledge extraction step is also executed on the value of 'rdfs:comment' collected with the factual knowledge. Figure 4.1 shows the steps involve in extracting meaningful phrases from acquired factual and contextual knowledge.

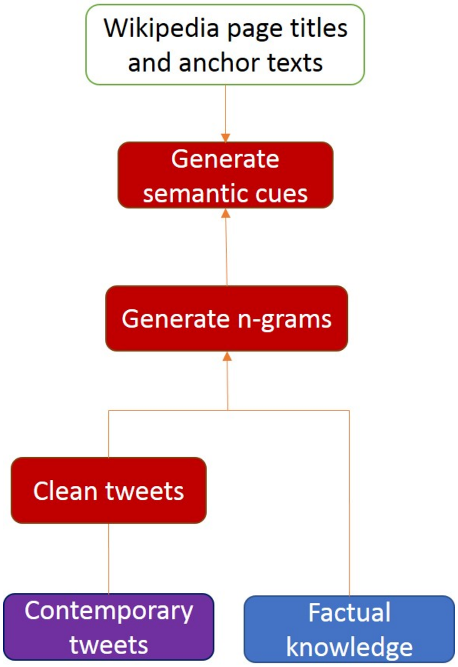


Figure 4.1: Extracting meaningful phrases from acquired knowledge

4.3.2 Entity Model Creation

The idea of the entity model is to capture the relevant knowledge of the entity and represent it in a way that can be leveraged to link the implicit mentions of entities. The entity model consists of three components: 1) factual knowledge, 2) contextual knowledge, and 3) time salience. The content for the first two components are obtained in the knowledge acquisition phrase, and the time salience of the entity is estimated with the number of hits that the corresponding page of Wikipedia gets during a predefined period. Figure 4.2 shows the three components of the entity model and sample content for each component for the movie *Gravity*.

The entity models of the multiple entities can share the same phrases. For example, the phrase ‘astronaut’ is relevant to entities such as *Gravity*, *Intersteller*, and *The Martian*; the phrase ’Matt Damon’ is relevant to both entities: *Intersteller* and *The Martian*. This observation led to integration of the entity models via common clues. The integrated entity models is called an ‘entity model network’. An entity model network reflects the topical relationship between the entities and quantifies the importance of each phrase to the associated

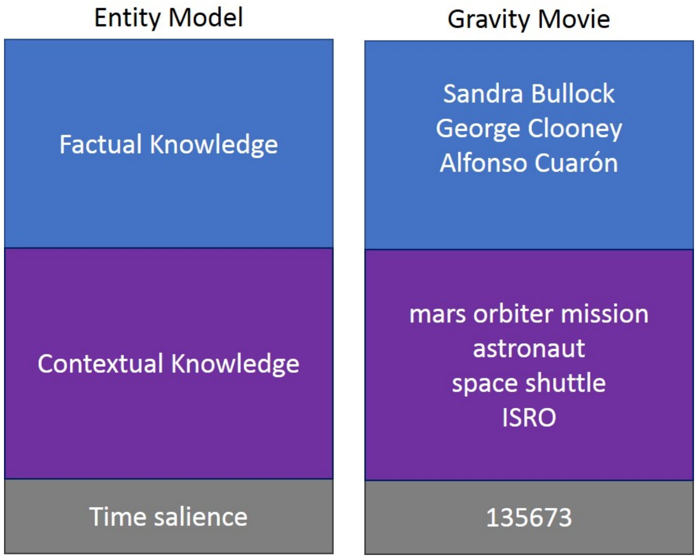


Figure 4.2: Three components of the entity model and sample content for the entity model of the movie *Gravity*

entities.

4.3.2.1 Entity Model Network

The entity models created for all the entities are integrated to generate the entity model network (EMN) as shown in Figure 4.3. There are two types of nodes, nodes that represent the entities and nodes that represent the clues. The entity nodes have the property 'time salience', which is the number of hits received by the Wikipedia page of the corresponding entity. The clue nodes have property 'specificity'; specificity of the clue node c_j is inspired by the inverse document frequency measure and is calculated as $\log \frac{|N|}{|N_{c_j}|}$, where $|N|$ is the number of entity nodes in the EMN and $|N_{c_j}|$ is the number of adjacent nodes to c_j. The 'frequency' property value of the edge between clue node c_j and entity node e_i is frequency of the clue present in the tweet collected for entity e_i. These properties associated with the entity nodes and clue nodes signify the importance of each clue to the entity (e.g., the higher the frequency value of the edge between c_j and e_i, the higher the importance of the clue c_j to the entity e_i).

Formally, an entity model network (EMN) is defined as a property graph $G_{EMN} = (V_e, V_c, E, \mu)$, where

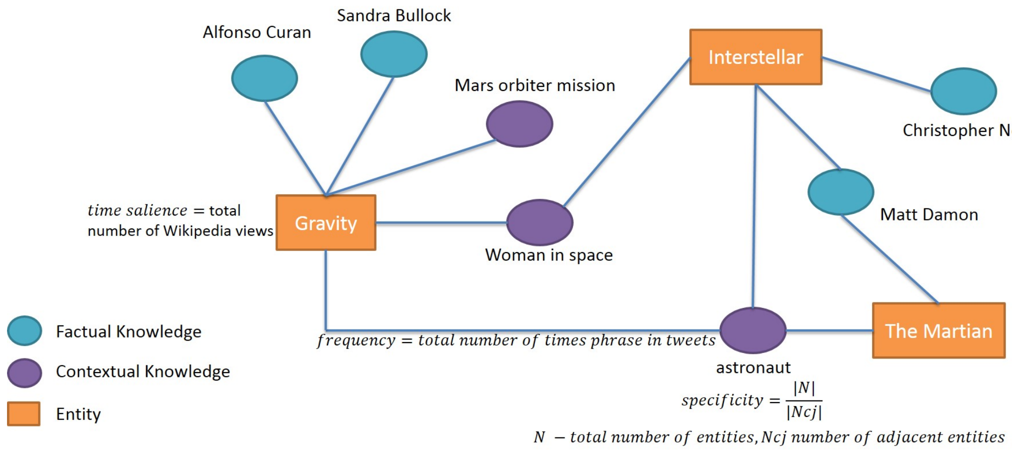

Sandra Bullock
Alfonso Curan
Mars orbiter mission
Interstellar
Christopher Nolan
time salience = total number of Wikipedia views
Gravity
Woman in space
Matt Damon
Factual Knowledge
Contextual Knowledge
Entity
frequency = total number of times phrase in tweets
astronaut
The Martian
specificity = |N| / |Ncj|
N – total number of entities, Ncj number of adjacent entities

Figure 4.3: Entity model network

V_e and V_c represent the nodes of two types, E represents the edges, and μ represents the property map. The edges are directed (i.e., $E \subseteq (V_c$ X $V_e))$, and μ maps the properties of nodes and edges as keys to values (i.e. $\mu : (V_e \cup V_c \cup E) X R \rightarrow S$), where R is a set of keys and S denotes values. V_e represents the entities and has the properties 'name' and 'time salience' as keys and their values as key/value pairs. V_c represents the clues and has the properties 'clue name' and 'specificity' as keys and their values as key/value pairs. The edges in the graph have the property 'frequency' and its value as key/value pair.

4.3.3 Detecting Tweets with Implicit Entities

The previous steps built the infrastructure that is needed to perform implicit entity linking. The next step is to identify the tweets with implicit entity mentions. This task is performed with a keyword based method. The keywords that indicate the type of the entities are selected to collect the tweets with entities of that type. For example, the keywords 'movie' and 'film' are used to collect the tweets with movies. If a collected tweet does not have an explicit mention of any entity of the required type, that tweet is considered to have an implicit entity mention. The presence of explicit entities in tweets can be identified with state-of-the-art entity linking tool.

4.3.4 Implicit Information Extraction - Linking Implicit Entities in Tweets

The entity model network is used to link the implicit entities in tweets. To understand how to use the EMN to perform implicit entity linking for a given tweet, it is useful to divide the task into two steps: 1) candidate selection and filtering, and 2) disambiguation.

4.3.4.1 Candidate Selection and Filtering

The objective of the candidate selection and filtering step is to prune the search space so that the disambiguation step does not have to evaluate all entities in EMN as candidates. In explicit entity linking this is usually done by looking for candidates with a given surface form. For example, if the surface form is ‘chicago’, it is possible to retrieve all entities that are indicated by term ‘chicago’ such as Chicago city, Chicago movie, Chicago Bulls, etc. from a dictionary. However, implicit entity manifestations do not have surface forms, hence there needs to be a different approach. The implemented approach identifies the phrases in the tweet using Wikipedia anchor text and page titles, and the terms that are not qualified as phrases are considered as unigrams. These identified phrases are referred to as ‘tweet clues’ and are denoted with $\mathcal{C}_t$. The candidate selection step takes these tweet clues and matches them with clue nodes in EMN. The entities that have at least one edge from matching clues are selected as the initial set of candidates.

Formally, given a set of tweet clues $\mathcal{C}_t$, the initial candidate entity set $\mathcal{E}_{\mathcal{IC}} = \{e_i | (c_j, e_i) \in E \text{ } and \text{ } c_j \in \mathcal{C}_t\}$.

The entities in the initial candidate set are scored based on the strength of evidences. The strength of evidence for entity $e_i \in \mathcal{E}_{\mathcal{IC}}$ (SC_{e_i}) is calculated in Equation (4.2).

$$SC_{e_i} = \sum_{c_j \in \mathcal{C}_t} \textit{specificity of } c_j * \textit{frequency of edge } (c_j, e_i) \tag{4.2}$$

The top k candidates based on these scores (denoted as E_c) are considered for the disambiguation step.

This procedure is demonstrated in Figure 4.4. Lets assume that the tweet *‘ISRO sends probe to Mars for less money than it takes Hollywood movie to send a woman to space’* is given as the input and the EMN

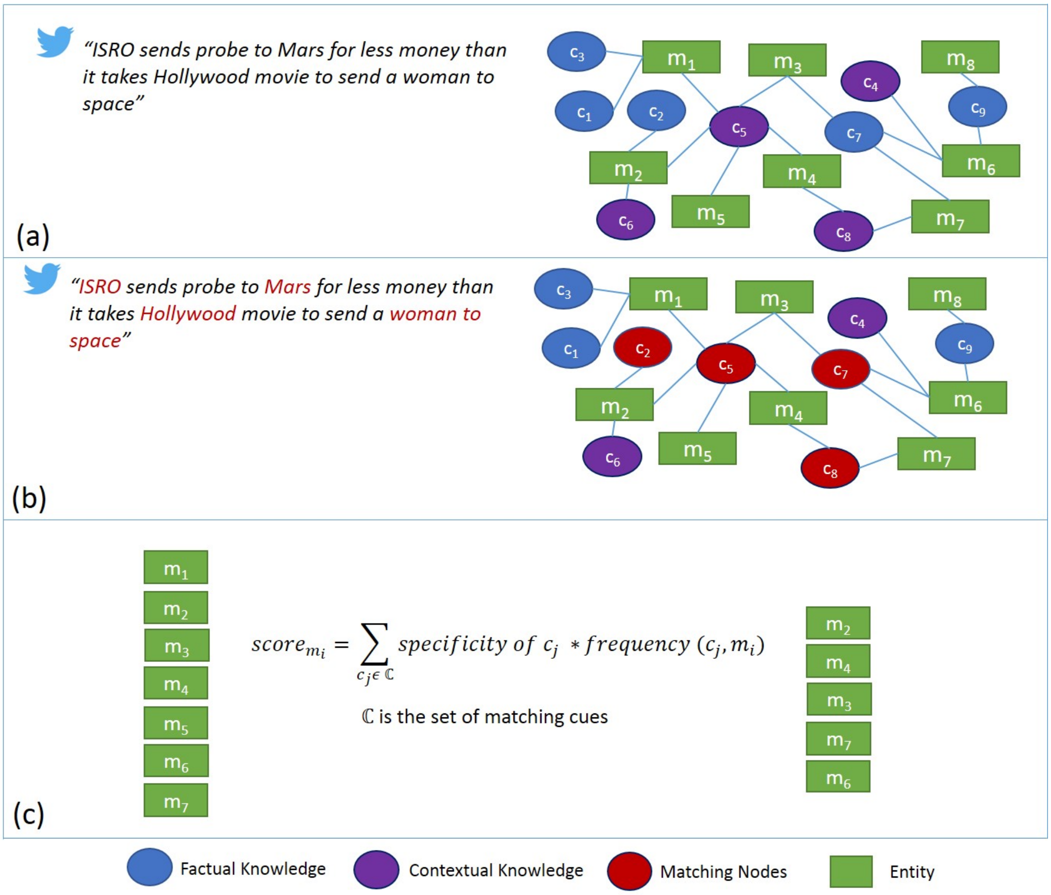

"ISRO sends probe to Mars for less money than it takes Hollywood movie to send a woman to space"
(a)
(b)
(c)
c_1 c_2 c_3 c_4 c_5 c_6 c_7 c_8 c_9
m_1 m_2 m_3 m_4 m_5 m_6 m_7 m_8
$score_{m_i} = \sum_{c_j \in \mathbb{C}} specificity\ of\ c_j * frequency\ (c_j, m_i)$
$\mathbb{C}$ is the set of matching cues
m_2 m_4 m_3 m_7 m_6
Factual Knowledge
Contextual Knowledge
Matching Nodes
Entity

Figure 4.4: Candidate selection and filtering

contains eight movies, and nine cues. The first step of the algorithm checks whether the n-grams in the tweet match with the cues in the EMN. According to this example, there are four matching cues. Hence, all the movies that have relationships with matching cues initially become candidates. The movies m_1-m_7 have relationships with the matching cues in the example (Figure 4.4(b)). In general, this candidate set can be huge. For example, there can be many movies that have relationship with the cue like 'Hollywood.' Hence, the next step scores the candidates based on the strength of the evidence according to Equation (4.2) (Figure 4.4(c)). The top k candidates after this scoring are given to disambiguation step as the input. In this example, the top-5 candidates are selected for disambiguation step.

4.3.4.2 Disambiguation

The objective of the disambiguation step is to sort the selected candidate entities such that the implicitly mentioned entity in a given tweet is at the top position of the ranked list. This is accomplished through a machine learned-ranking model based on the pairwise approach: all pairs of selected candidate entities (along with a feature set) are taken as input, and the model approximates the ranking as a classification problem that tells which of the entities in the pair is better than the other.

The feature set of a candidate entity consists of its similarity to the tweet and its time salience w.r.t. the time salience of other candidate entities.

The similarity between the candidate entity and the tweet can be calculated via their vector representations. The vector representation of the candidate entity e_i is obtained via its incoming connections from other nodes. It is denoted as e_{i_v} and defined as $e_{i_v} =< v_1, v_2, ..., v_n >$, where $v_j = \textit{specificity of } c_j * \textit{frequency of edge } (c_j, e_i)$ for all $(c_j, e_i) \in E$. The vector representation of the tweet is created using tweet clues. The similarity between the candidate entity and the tweet is calculated by the cosine similarity of these vectors.

The time salience of the candidate entity e_i is normalized w.r.t the time salience of other candidate entities in E_c as:

$$\frac{\textit{time salience of } e_i}{\sum_{e \in E_c} \textit{time salience of } e} \tag{4.3}$$

The disambiguation step trained a SVM^{rank} model to solve the ranking problem. It used linear kernel, 0.01 as the trade-off between training error and margin, and total number of swapped pairs summed over all queries as the loss function. SVM^{rank} has been shown to perform well in similar ranking problems; specifically, it is able to provide best performance in ranking the top concept [Meij et al. 2012], which suits the characteristics of entity linking problem.

4.4 Evaluation

The evaluation was conducted in two domains, namely, Movie and Book. There is no gold standard dataset available to evaluate this task. Hence, an evaluation dataset has been created for the two domains. The evaluation is focused on answering three questions.

- How effective is the proposed approach in linking implicit entities?
- How important is the contextual knowledge in linking implicit entities?
- What is the value added by linking implicit entities?

4.4.1 Dataset Preparation

In order to prepare datasets for the evaluation, we collected tweets with 'movie' and 'film' as keywords for the Movie domain and 'book' and 'novel' as keywords for the Book domain. This dataset was collected during August 2014. The keyword-based data collection can select the tweets with explicit mention of entities, implicit mention of entities, and tweets with no entity mentions at all (for example, the tweet *'lets have a movie night today'* has the keyword 'movie', but does not mention any particular movie either explicitly or implicitly). The collected tweets were manually annotated by two individuals with 'explicit', 'implicit', and 'NIL' labels for Movies and Books. The annotations agreed upon by both individuals are included in the evaluation dataset. Table 4.1 shows the important characteristics of the annotated dataset and it is available at `https://goo.gl/jrwpeo`.

As shown in Table 4.1, there were 391 tweets annotated as 'explicit' for the movie domain, and these tweets had 107 mentions of distinct movie entities. There were 207 tweets with implicit entities and 54 mentions of distinct movies. Each tweet had between 16 and 18 words.

To perform implicit entity linking on the aforementioned evaluation dataset, the EMN was created for the 31st of July 2014. In order to identify the entities of interest during this time period, the most recent 15,000 tweets upto 31st of July 2014 were collected for each domain using its type labels as the keywords

Domain	Annotation	Tweets	Entities	Avg. length
Movies	explicit	391	107	16.5 words
	implicit	207	54	18 words
	NIL	117	0	16.4 words
Books	explicit	200	24	18.5 words
	implicit	190	53	18.5 words
	NIL	70	0	17.5 words

Table 4.1: Evaluation dataset statistics. Describes, per domain, the total number of tweets per annotation type (explicit, implicit, NIL), number of distinct entities annotated, and average tweet length.

(e.g., 'movie' and 'film' for movie domain) and a simple spotting mechanism was applied to identify the explicit entity mentions. The spotting mechanism collected the labels of the domain entities from DBpedia (i.e. rdfs:label value of the instances of type Film) and then checked for their presence in the collected dataset for the domain. If the label was found within the tweets, the corresponding entity was selected to be added to the EMN. For each entity selected: a) the most recent 1,000 tweets were collected that explicitly mention that entity to extract their contextual knowledge, b) factual knowledge about the entity was extracted from the DBpedia version created with May 2014 Wikipedia dumps, and c) page hit counts of Wikipedia pages for the month of July 2014 were obtained to estimate the time salience. This process provides the required knowledge to build the entity model network. Recall that the factual knowledge acquisition step ranks the relationship for each domain and selects the top m relationships (Equation (4.1)). It is observed that the entity linking solution performs well when $m = 15$; hence, the factual knowledge component consists of knowledge extracted with the top 15 relationships. A detailed study on the consequences of varying this parameter is presented later in this section. The created EMNs for the Movie and Book domains had 617 entities and 102 entities respectively.

Top-k value	Candidate Selection Recall (%)	Disambiguation Accuracy (%)
5	78.74	57.00
10	85.50	57.97
15	88.40	60.86
20	88.89	61.35
25	90.33	60.97
30	90.82	60.86
35	91.78	61.83

Table 4.2: Implicit entity linking performance on Movie domain

Top-k value	Candidate Selection Recall (%)	Disambiguation Accuracy (%)
5	87.36	58.94
10	92.10	60.00
15	92.63	60.52
20	93.15	61.57
25	94.73	61.05
30	94.73	61.05
35	94.73	61.05

Table 4.3: Implicit entity linking performance on the Book domain

4.4.2 Implicit Entity Linking Evaluation

This section evaluates the implicit entity linking task in isolation. It shows the results on both the candidate selection and filtering, and disambiguation steps. The candidate selection and filtering is evaluated as the proportion of tweets that had the correct entity within the top k selected candidates for that tweet (denoted as Candidate Selection Recall). The disambiguation step is evaluated with 5-fold cross validation and the results are reported as a proportion of the tweets in the evaluation dataset that had the correct annotation at the top position. The experiment is conducted by varying k between 5 and 35 and results are shown in Table 4.2 and Table 4.3.

As shown in Table 4.2 and Table 4.3, increasing the top-k candidates increases the accuracy from 58.94% to 61.57% (books) and from 57% to 61.83% (movies) for k=5, 10, 15, 20, 25, 30, 35 with more significant increases in smaller k values and coming to a near plateau around k=20. The small dip exhibited in the disambiguation graph after k=20 may be due to following reason. The candidates are selected based on the strength of the evidence calculated based on weighted content overlap between the entity model and the tweet. The time salience feature is not considered in the candidate selection step. However, time salience is considered in the disambiguation step. So with the incrementation of k, it is possible that new candidates were included with high time salience and they get ranked top due to their dominance in the time salience value.

Qualitative Error Analysis The error analysis on implicit entity linking shows that errors are fourfold: 1) errors due to lack of contextual knowledge of the entity, 2) errors due to novel entities, 3) errors due to cold start of entities and topics, and 4) errors due to multiple implicit entities in the same tweet. Table 4.4 shows an example tweet for each error type.

The first two tweets in the table are annotated with the movie *White Bird in a Blizzard*. There were only 46 tweets for this movie. Hence, the contextual knowledge component of the entity model did not provide strong evidence in the disambiguation step. The tweets with the second error type are annotated with the movie *Deepwater Horizon*. The Wikipedia page for this movie was created on September 2014, hence it was

Error	**Tweet**
lack of context	*‘That Movie Where Shailene Woodley Has Her First Nude Scene? The Trailer Is RIGHT HERE!: No one can say Shailene Woodley isn’t brave!’*
	‘Scrolling down my timeline I see Jamie Dornan nude photo-shoot and Shailene Woodley naked movie scenes.. it’s definitely time to sleep’
novel entities	*“‘hey, what’s wrawng widdis goose?” RT @TIME: Mark Wahlberg could be starring in a movie about the BP oil spill http://ti.me/1oZh55V’*
	‘Can Mark Wahlberg Transform Focus Of BP Disaster Movie? http://ift.tt/1oYEGcN moviesteem http://bit.ly/1iaKKIF #movie’
cold starts	*‘Video: George R.R. Martin’s Children’s Book Gets Re-release http://bit.ly/1qNNH5r’*
	‘A kid version of Game of Thrones? George RR Martins children’s book reads more like a Hans Christian Anderson tale’
multiple entities	*‘That moment when you realize that hazel grace and Augustus are brother and sister in one movie and in love battling cancer..’*
	‘Ansel Elgort & Shailene Woodley were brother and sister in one movie, then two people in love with cancer in another. I don’t like that’

Table 4.4: Example tweets for each error type.

not available to the EMN created for evaluation. This is known as an emerging entity discovery problem and requires separate attention as in the task of explicit entity linking [Hoffart et al. 2014]. A few entities and topics emerged among Twitter users only after 31 July 2014. These entities and topics were not present in our EMN. One of them is the republication of George R.R. Martin's book *The Ice Dragon*, which emerged in early August 2014 resulting tweets about the book. Two such tweets are shown in the fifth and sixth rows in Table 4.4. Both the entity and the topic were not known to the EMN, hence it couldn't link the tweet to the book *The Ice Dragon*. This problem can be solved by implementing an evolution mechanism for the EMN. Lastly, a couple of tweets in the dataset had two implicit entities. The last two tweets in table are annotated with the movies *The Fault in Our Stars* and *Divergent*. Since the developed solution links only one entity per tweet, there are two choices available; either remove the tweets with two mentions or add it once for each mention. The evaluation did the latter to preserve the characteristics of the tweets. However, not surprisingly, both tweets were annotated with the same movie (*The Fault in Our Stars*) resulting in an incorrect annotation each. Although this is a limitation of the solution, this phenomenon is not a frequent occurrence as the dataset had only 6 (=1.5%) tweets with two implicit entity mentions.

4.4.3 Importance of Contextual Knowledge

One of the major components of the entity model is the contextual knowledge of the entity. This section evaluates its contribution to the proposed implicit entity linking solution by comparing the results obtained by EMN created with contextual knowledge and EMN created without contextual knowledge. As in Section 4.4.2, it is evaluated for both the candidate selection and filtering, and disambiguation steps.

Table 4.5 shows the results of this experiment. As shown in Table 4.5, the contextual knowledge contributes to both the candidate selection and filtering, and disambiguation steps on both domains. Quantitatively, the recall value of the first step increased by 14% and 19% while the accuracy of the second step increased by 15% and 18% for movies and books, respectively. This is due to the fact that people do not necessarily use the associated entities when referring to entities implicitly, but rely on other clues. For example, consider tweets *"'My name was Salmon, like the fish; first name, Susie." Great book!'* and *"2 actors playing*

Step	Domain	Without ctx	With ctx
Candidate Selection Recall	Movie	77.29%	90.33%
	Book	76.84%	94.73%
Disambiguation	Movie	51.7%	60.97%
	Book	50.00%	61.05%

Table 4.5: Contribution of the contextual knowledge component of entity model.

brother and sister then plot twist new movie, but they have cancer and love each other." The first tweet has an implicit mention of the book *The Lovely Bones* and the second tweet has implicit mentions of the movies *The Fault in Our Stars* and *Divergent.* However, none of them contain any entities associated (e.g., author, publisher, actor, director) with the book or to the movie. Hence, the factual knowledge component in the entity model fell short in these tweets. The contextual knowledge component fills in the gaps since it can build the association between the clues indicated in tweets and the respective entities.

4.4.4 Value Addition to Standard Entity Linking Task

The real-world datasets collected via keywords contain explicit and implicit entity mentions as well as tweets with no entity mentions. This experiment assesses the impact of implicit entity linking in such datasets. To create a dataset for this experiment the following steps were followed:

- Select 40% of the tweets from the dataset with implicit entities as the test dataset. We use the rest to train the ranking model.
- Mix the selected test dataset with tweets containing explicit entity mentions by preserving the explicit:implicit ratio. This ratio is 4:1 in the Movie domain and 5:2 in Book domain.
- Add 25% of tweets that have no mention of an entity to account for NIL mentions.

The experiment setup used three well-known entity linking solutions with the following configurations: 1) DBpedia Spotlight [Daiber et al. 2013] (confidence=0.5, support=20), 2) TagMe [Ferragina and Scaiella

	DBpedia Spotlight		**TagMe**		**Zemanta**	
	Movies	Books	Movies	Books	Movies	Books
F1 (EL)	0.18	0.44	0.24	0.19	0.32	0.17
F1 (EL+IEL)	0.34	0.54	0.30	0.34	0.39	0.37

Table 4.6: EL and IEL combined performance

2010] (ρ=0.5), and 3) Zemanta.[4] It first annotates the prepared Movie dataset using DBpedia Spotlight. The tweets that are not annotated with movies are assumed to have implicit entity mentions and are sent to the implicit entity linking solution. The same exercise is repeated for TagMe and Zemanta. This experiment is conducted for the Book dataset.

Table 4.6 shows the results of this experiment using F1 measure. The precision (P) and recall (R) values for F1 measure are calculated as follows:

$$P = \frac{\textit{total correctly annotated tweets}}{\textit{total tweets annotated with entity}} \quad \text{and} \quad R = \frac{\textit{total correctly annotated tweets}}{\textit{total tweets with entity}},$$

These results demonstrate the value of adding IEL as a post-step to EL.

4.4.5 The Impact to the Implicit Entity Linking by Varying the Number of Relationships Considered to Collect Factual Knowledge

As mentioned earlier, there is another parameter that can impact the entity linking performance, i.e., the number of relationships selected for acquiring factual knowledge about the entities. The relevancy of the relationships to the domain of interest is calculated by Equation (4.1). An experiment was conducted to select the number of best top relationships that should be considered to acquire factual knowledge. The number of relationships selected were varied between 5 and 20, and then creating the EMN using the factual knowledge extracted with the selected relationships, and evaluating the performance of the entity linking task. As shown Figure 4.5, entity linking has the best performance when the top 10 relationships are selected. It starts to have

[4] http://www.zemanta.com

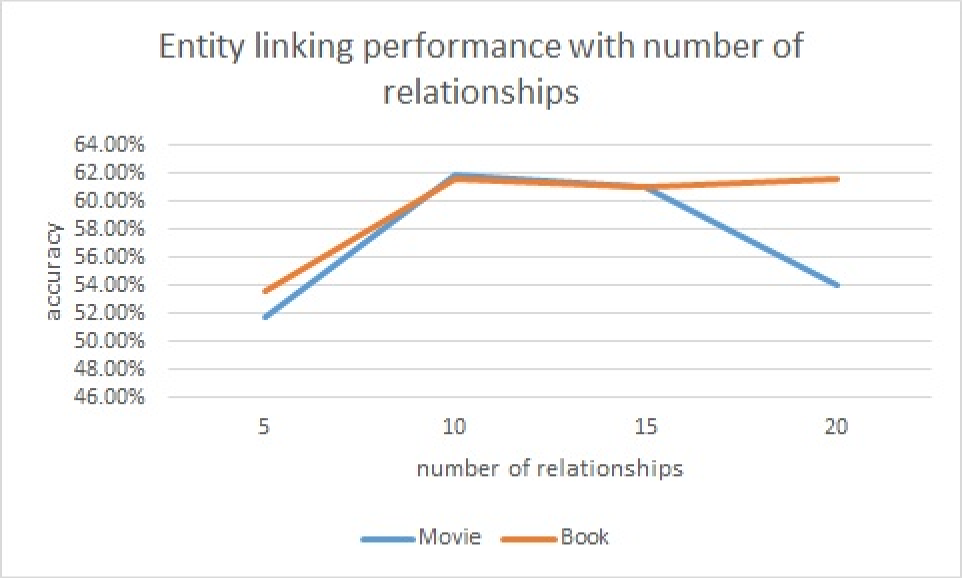


Figure 4.5: Entity linking performance by varying the number of relationships

a negative impact when adding more than 15 relationships. This is likely due to the inclusion knowledge that isn't particularly relevant. The ranked relationships for movies and books are shown in Table 4.7.[5] As shown, the popular relationships like *starring* and *director* for movies and *author* and *publisher* for books are ranked between 5_{th} and 10_{th} position. This explains the results shown in Figure 4.5.

4.5 Conclusion

This chapter studied the implicit entity linking problem in tweets. Twitter is a very popular communication platform which emerged in recent years with the advent of the social media. The data generated by Twitter users are used in many applications. These applications are required to identify the entities mentioned in the tweets to achieve their ultimate goal. Hence, the failure to identify the implicitly mentioned entities in tweet can negatively impact the output of these applications.

Twitter users assume a shared knowledge with the audience that lets them to constrain the message to 140 characters, and yet make it understandable by the target audience. It is observed that, a solution developed to

[5] All relationships are prefixed with `http://dbpedia.org/ontology`. This information is omitted to improve the readability.

Rank	Movie	Book
1	cinematography	nonFictionSubject
2	editing	coverArtist
3	openingFilm	translator
4	musicComposer	mediaType
5	closingFilm	literaryGenre
6	distributor	illustrator
7	starring	author
8	director	publisher
9	basedOn	notableWork
10	writer	series
11	narrator	basedOn
12	film	language
13	language	lastAppearance
14	producer	subsequentWork
15	lastAppearance	previousWork
16	notableWork	country
17	country	currentProduction
18	photographer	notableIdea
19	knownFor	closingFilm
20	portrayer	knownFor

Table 4.7: Top ranked relationships for movie and book domains.

link the implicit entities in tweet should possesses this shared knowledge. This chapter presented the details of the developed solution which harnesses the structured knowledge bases and daily communications among Twitter users to acquire the required knowledge and model the entities of interest. These entity models were used to perform the implicit entity linking, which is formulated as a ranking problem.

The developed solution linked the implicit entities with an accuracy of 61%. It also showed the value of contextual knowledge extracted from daily communications and the overall value of linking implicit entities with explicit entities.

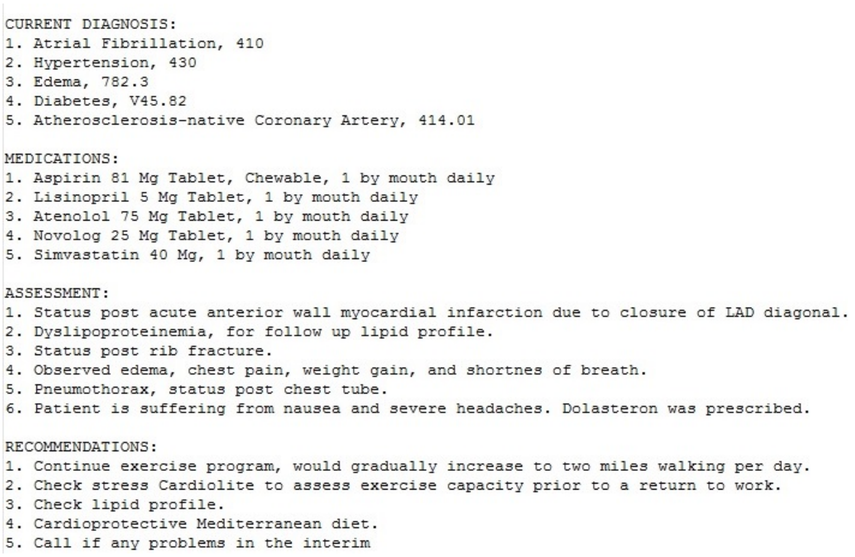
5

Extracting Implicit Relationships between Entities in Clinical Documents

5.1 Overview

Clinical narratives are full of relationships between entities, but they rarely express them explicitly. Clinical documents consist of multiple sections, including patient health history, family history, current diagnosis, current medications, and clinical recommendations. These sections contain entities of the type disorders, symptoms, medications, and procedures. These entities have clinically relevant relationships between them. The symptoms are *caused by* the disorders, the medications are *prescribed for* the symptoms/disorders, the procedures are *performed to treat or diagnose* disorders. These relationships are very important to understand the health status of a patient. With the recent development in healthcare policies in the United States, there is a huge demand to develop solutions that leverage Electronic Medical Records (EMR) of patients to perform predictive analytics. Such tasks require understanding the relationships that exist between entities in clinical documents.

Consider the example clinical narrative shown in Figure 5.1, it has four sections, namely current diagnosis, medications, assessment, and recommendations. Each section has listed down relevant clinical entities.



CURRENT DIAGNOSIS:
1. Atrial Fibrillation, 410
2. Hypertension, 430
3. Edema, 782.3
4. Diabetes, V45.82
5. Atherosclerosis-native Coronary Artery, 414.01

MEDICATIONS:
1. Aspirin 81 Mg Tablet, Chewable, 1 by mouth daily
2. Lisinopril 5 Mg Tablet, 1 by mouth daily
3. Atenolol 75 Mg Tablet, 1 by mouth daily
4. Novolog 25 Mg Tablet, 1 by mouth daily
5. Simvastatin 40 Mg, 1 by mouth daily

ASSESSMENT:
1. Status post acute anterior wall myocardial infarction due to closure of LAD diagonal.
2. Dyslipoproteinemia, for follow up lipid profile.
3. Status post rib fracture.
4. Observed edema, chest pain, weight gain, and shortnes of breath.
5. Pneumothorax, status post chest tube.
6. Patient is suffering from nausea and severe headaches. Dolasteron was prescribed.

RECOMMENDATIONS:
1. Continue exercise program, would gradually increase to two miles walking per day.
2. Check stress Cardiolite to assess exercise capacity prior to a return to work.
3. Check lipid profile.
4. Cardioprotective Mediterranean diet.
5. Call if any problems in the interim

Figure 5.1: Electronic Medical Record

However, one would not be able to know the medications prescribed for each disorder or the relevant symptoms for each disorder in the document by reading it. A medical professional reading a clinical narrative can make all these implicit relationships explicit by leveraging his domain knowledge. He would know that atenolol and lisinopril is prescribed to treat hypertension, insulin is prescribed to treat diabetes, atrial fibrillation and hypertension cause chest pain etc. These relationships are visualized in Figure 5.2.

The algorithms developed to extract relationships between various entities from text either need them to be mentioned explicitly in the text or need to be given in a knowledge base that contains these relationships. If the relationship is mentioned explicitly, these algorithms use pattern-based [Mooney and Bunescu 2005] [Hearst 1992] [Aone and Ramos-Santacruz 2000], dictionary-based [Chun et al. 2006], and verb-based [Sharma et al. 2010] approaches. Another idea to extract relationships between entities in text is to use prior knowledge about the relationships between entities [Mintz et al. 2009] [Abulaish and Dey 2007]. By definition, implicit relationships are not expressed in text explicitly; hence, a comprehensive knowledge base of clinical knowledge is a necessity to elicit relationships between clinical entities. Creation of clinical knowledge bases is a well-worked out task. As a result, there are many clinical knowledge bases available and the Unified Medical Language System (UMLS) is a resource created by integrating such knowledge bases.

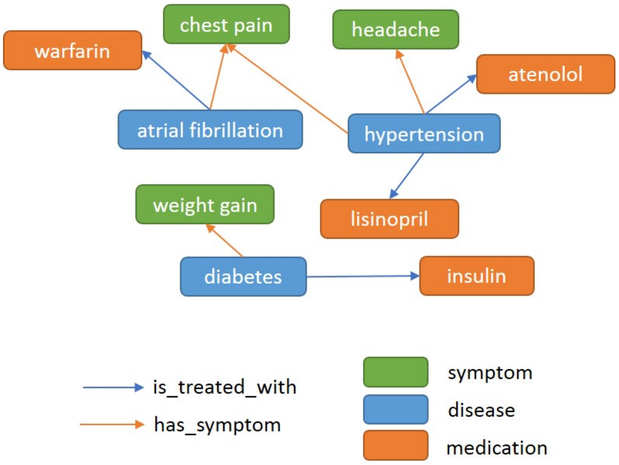


Figure 5.2: Relationships between clinical entities

UMLS has rich coverage of entities and the hierarchical relationships between them. However, it falls short of establishing non-hierarchical relationships between entities. This poses great challenge in eliciting implicit relationships between entities in clinical documents. Hence, this chapter addresses the problem of a lack of relevant non-hierarchical knowledge about the entities in the clinical knowledge bases and develop algorithm that efficiently acquire non-hierarchical knowledge.

Traditionally people used methods such as interviewing domain experts, finding facts from literature, and validating known facts with existing data/use cases to collect the knowledge required to build knowledge bases. The knowledge collected by these methods is observed to be subjective, ambiguous, and incomplete. This is particularly true in a domain like healthcare, since an individual's knowledge significantly depends on his/her experience. In addition, this is a tedious task; for example, imagine that knowledge base has 50 disorders and 100 symptoms, this calls for inquiring about 5,000 (50*100) relationships from domain experts. This is almost a impossible task given the limited availability of the domain experts in the healthcare domain.

This chapter focuses on a knowledge acquisition solution that can effectively acquire knowledge on relationship between clinical entities by formulating hypotheses with high accuracy. It analyses the clinical narratives of the patients, establish the known relationship between entities to the given knowledge base, identifies the relationships that are unknown to the knowledge base, and formulates questions with high accuracy whose answers can rectify the gaps in the knowledge base and populate implicit relationships. The evaluation demonstrates the efficiency of the knowledge acquisition process and how that helps in populating the implicit relationships between disorders and symptoms in clinical narratives.

5.2 Related Work

A comprehensive ontology evolution framework proposed in [Zablith 2009] uses Scarlet [Sabou et al. 2008] to find the relationships among the concepts. Scarlet [Sabou et al. 2008] uses multiple rules to derive hierarchical relationships as well as domain specific relationships. While these rules are capable of deriving hierarchical relationships by integrating multiple ontologies, they cannot derive non-hierarchical relationships. Scarlet depends on the availability of such relationships in existing knowledge bases.

Ontology learning from a text corpus is a well-known task. Algorithms developed to perform this task use NLP and ML-based techniques to identify the domain entities and hierarchical and non-hierarchical relationships. These techniques rely on named entity identification methods [Ciaramita et al. 2005], predefined linguistic patterns [Ciaramita et al. 2005] [Sánchez and Moreno 2006] [Xiao et al. 2009], lexical syntactic properties of the free text (like the frequency of words appearing together [Ciaramita et al. 2005] [Kavalec et al. 2004] [Xiao et al. 2009], the position of the words [Faure and Nédellec 1998], and the frequency of verbs appearing with the lexical terms [Kavalec et al. 2004] [Schutz and Buitelaar 2005]).

Understanding causal relationships is a fundamental requirement for text comprehension. People have developed techniques to mine the causal relationships from text and the majority of these techniques follow the same methods mentioned above. [Blanco et al. 2008] [Khoo et al. 1999] [Girju et al. 2002] use syntactic patterns and [Do et al. 2011] use a co-occurrence based method to identify causal relationships. [Mulkar-Mehta et al. 2011] proposes a method to identify causal relationships by using part-of relationships. It claims that causal relationships can be identified using fine-grained events.

The solution developed in this chapter is different from the above methods as it focuses on extracting implicit relationships among entities in clinical narratives. The inputs are seed knowledge base with known relationships, large collection of EMRs, and the output is updated knowledge base with unknown relationships. The updated knowledge base is used to extract the implicit relationships between entities in clinical documents.

5.3 Implicit Relationship Extraction from Clinical Narratives

The main hindrance in extracting implicit relationships from the clinical narrative is the unavailability of a comprehensive background knowledge base. Hence, the objective of the developed solution is to develop an efficient technique to acquire knowledge on non-hierarchical relationships between clinical entities. The developed solution takes entities of interest as the input, acquires the knowledge on hierarchical and known non-hierarchical relationships between entities, models the entities using their hierarchical and non-hierarchical relationships, analyses the clinical narratives to detect implicit relationships not present in the knowledge base, and generates the questions that fill the gaps in the knowledge base and populate unknown implicit relationships to the given knowledge base in clinical narratives.

This dissertation uses the ontology of perception IntellegO [Henson et al. 2011] to formally model the non-hierarchical relationships between entities and reason over them to generate hypotheses about implicit relationships. The following section provides an introduction to IntellegO.

5.3.1 Ontology of Perception (IntellegO)

Perception is the process of interpreting observations of the environment to derive situational awareness; in other words, the process of translating low-level observations into high-level abstractions. IntellegO is an ontology that provides formal semantics of machine perception (perceptual reasoning) by defining the informational processes involved in translating observations into abstractions.

Diagnosis is a function of perception. Medical professionals derive disorders (abstractions) by examining symptoms (low level signals). Clinical narratives implicitly contain the knowledge involved in this perceptual process. The semantics formalized with the IntellegO ontology and the perceptual reasoning mechanism nicely aligns with the informational process in clinical narratives.

The following scenario demonstrates the suitability of IntellegO to represent the knowledge in the clinical domain. IntellegO has defined classes *'intellego:entity'*, *'intellego:quality'*, *'intellego:percept'*, and *'intellego:explanation'* and the relationship *'intellego:inheres-in'*. In general, *'intellego:entity'* is an object or event in the real world and *'intellego:quality'* is an inherent property of an *'intellego:entity'*. *'intellego:inheres-in'* is the relationship between *'intellego:quality'* and *'intellego:entity'*. Let's take the disorder *hypertension* and an associated symptom, *chest pain*. A disorder is an entity and associated symptoms are qualities of the disorder. Hence, *hypertension* is an *'intellego:entity'* and *chest pain* is an *'intellego:quality'*. The *'intellego:quality' chest pain 'intellego:inheres-in' 'intellego:entity' hypertension*.

The perception process begins by observing a few qualities. From these observations, it derives entities which can explain all observed qualities. Assume that *chest pain* is observed in a clinical narrative. *chest pain* can be explained by the presence of *hypertension*[1]. The set of **observed** *'intellego:quality'* (i.e., *chest pain*) are members of the class *'intellego:percept'* and *'intellego:entity'* (e.g., *hypertension*) which can **explain** the *'intellego:percept'* are members of the class *'intellego:explanation'*. *'intellego:explanation'* and *'intellego:percept'* are sub-classes of *'intellego:entity'* and *'intellego:quality'* respectively.

As illustrated in this section, the concepts from the IntellegO ontology map to the concepts in the health care domain as[2]: *'intellego:entity'* to `DISORDER`, *'intellego:quality'* to `SYMPTOM`, *'intellego:percept'* to `OBSERVED_SYMPTOM`, *'intellego:explanation'* to `EXPLANATORY_DISORDER`, and *'intellego:inheres-in'* to `IS_SYMPTOM_OF` relationship. These terms will be used instead of IntellegO classes to improve the readability.

[1] Note that there may be multiple entities that can explain the observed qualities (e.g., *chest pain* can be explained by *hypertension*, *cardiomyopathy*, *coronary artery disease* and a host of other disorders), but for simplicity, we assume *chest pain* can be explained only by *hypertension*.

[2] We use the uppercase term to distinguish between the domain entity and the corresponding IntellegO class

5.3.2 Knowledge Acquisition

The knowledge acquisition step of the solution obtains the hierarchical relationships of entities of interest and the seed set of non-hierarchical relationships between entities. This knowledge can be found in UMLS and Web resources with domain experts' help.

UMLS describes hierarchical relationships between concepts. As explained in Section 3.3.1, the UMLS vocabulary is serialized as a relational database. The relational table *MRHIER* contains hierarchical relationships between entities. Each entity in UMLS has two types of identifiers: 1) *AUI* - Atom Unique Identifier that represents an entity with multiple surface forms ('dyspnea' and 'shortness of breath' refer to same entity) and is defined in multiple vocabularies. Each such definition in each vocabulary is assigned a unique identifier called AUI. 2) *CUI* - Concept Unique Identifier which is a unique ID assigned to each entity based on its meaning. In other words, this merges all surface forms of the same entity to provide it a unique identity. The hierarchical relationships in *MRHIER* are defined over *AUI*s. The columns named *AUI*, *PAUI*, and *PTR* in the *MRHIER* table provide hierarchical knowledge of the entities. The *PAUI* column contains the immediate parents of the entities listed in *AUI* column. The *PTR* column contains a path of the hierarchy written as a string delimited with periods. For instance, the entity with *AUI* 'A0031972' has *PTR* 'A0434168.A2367943.A18456972.A0135391.A0088829'. The leftmost *AUI* in the path is the most generic entity and the rightmost *AUI* is the most specific entity in the hierarchy for 'A0031972'. This data in UMLS can be used to build the hierarchy for each entity of interest.

The seed set of non-hierarchical relationships can be populated in a semi-automated manner. For example, there are credible resources on the Web that list the relationships between entities. The following Web resources were used to identify a seed set of non-hierarchical relationships between entities along with the guidance from domain experts.

- NLM (`www.nlm.nih.gov`)
- PubMed (`http://www.ncbi.nlm.nih.gov/pubmed`)
- WebMD (`www.webmd.com`)

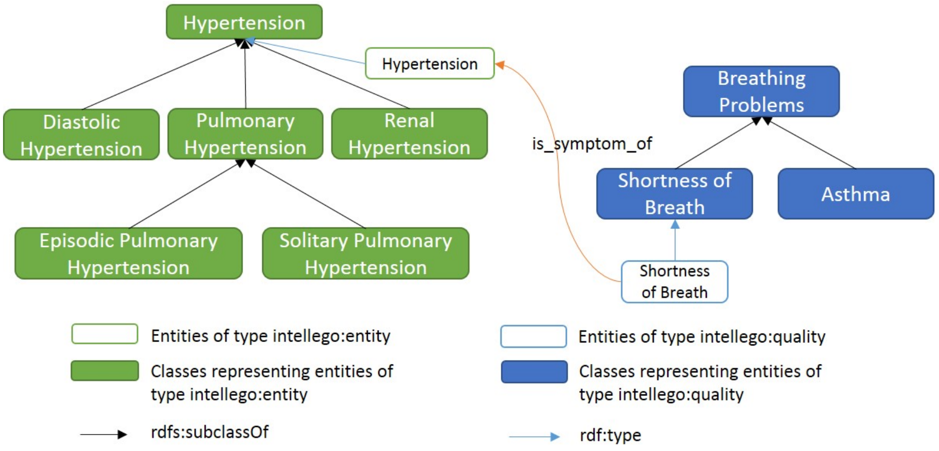

Hypertension
Hypertension
Breathing Problems
Diastolic Hypertension
Pulmonary Hypertension
Renal Hypertension
is_symptom_of
Shortness of Breath
Asthma
Episodic Pulmonary Hypertension
Solitary Pulmonary Hypertension
Shortness of Breath
Entities of type intellego:entity
Classes representing entities of type intellego:entity
rdfs:subclassOf
Entities of type intellego:quality
Classes representing entities of type intellego:quality
rdf:type

Figure 5.3: Entity model with hierarchical and non-hierarchical relationships

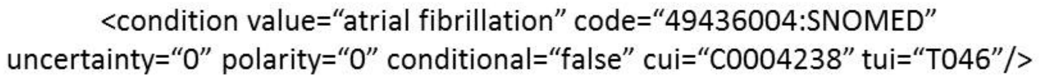
- Cleveland Clinic (`www.clevelandclinic.org`)
- Wikipedia (`www.en.wikipedia.org`)
- Mayo Clinic (`www.mayoclinic.com`)
- Healthline (`http://www.healthline.com`)

5.3.3 Knowledge Modelling

The knowledge modeling step involve modeling the clinical entities with known non-hierarchical relationships and hierarchical relationships. Non-hierarchical relationships are modelled with the IntellegO ontology as described in Section 5.3.1. The hierarchical relationships are represented with the ‘rdfs:subclassOf’ relationship. Figure 5.3 shows a snippet of the entity model.

5.3.4 Detecting Implicit Relationship between Disorders and Symptoms

The next step is to detect implicit relationships in clinical narratives. In order to do this, this dissertation assumes that clinical narratives are consistent. The notion of consistency in clinical narratives is that ‘symptoms appearing in the document are accounted for by the disorders in it.’ Hence, all symptoms in a clinical narrative should be linked to a disorder. The relationships between symptom and disorder are causal in nature, i.e., the presence of disorder cause the presence of symptoms. The relationships specified in entity models



```
<condition value="atrial fibrillation" code="49436004:SNOMED"
uncertainty="0" polarity="0" conditional="false" cui="C0004238" tui="T046"/>
```

Figure 5.4: XML element describes *Atrial Fibrillation* in cTAKES output.

are used to establish relationships between symptoms and disorders.

The first step towards detecting implicit relationship is to find the symptoms and disorders mentioned in clinical narratives. This is accomplished by performing entity linking. There are multiple tools which are capable of performing entity linking in clinical documents. Among them, cTAKES [Savova et al. 2010] has been chosen to perform the entity linking due to following reasons:

- It has been developed for the healthcare domain.
- It has a high accuracy.
- It annotates concepts using UMLS vocabulary.
- It is capable of extracting rich set of information about the concepts (e.g., if the concept is a disorder or a symptom, it is capable of identifying properties such as whether a document mentions the presence or absence of disorder/symptom, whether it is a current disorder/symptom or part of the history, determine the section of the EMR document in which the concept appears, and semantic type of the concept).
- It outputs an XML file which is easily processed by machines.
- It is an open source tool.

Figure 5.4 shows an example element from an XML document generated by cTAKES which explains a disorder found in a clinical narrative.

The annotated symptoms and disorders are represented as `OBSERVED_SYMPTOM` and `EXPLANATORY_DISORDER` respectively. This allows us to use the knowledge in entity models to populate `IS_SYMPTOM_OF` relationships between symptoms and disorders. Populating `IS_SYMPTOM_OF` relationships leveraging the knowledge that exists in entity models is a straightforward task. However, there can still be symptoms that are not accounted for by a disorder after this step, and they are termed as an 'unexplained symptom.'

Formally, the algorithm performs perceptual reasoning to identify unexplained symptoms. This is accomplished by finding all possible symptoms that can be explained by the disorders present in the document (i.e., EXPLANATORY_DISORDER) and comparing this symptom set with the OBSERVED_SYMPTOM set. If there is a symptom in OBSERVED_SYMPTOM which is not in all possible symptom sets, it is regarded as an unexplained symptom. In order to perform this task, this dissertation adds the class *'intellego:coverage'* to IntellegO ontology.

'intellego:coverage' can be defined as the aggregation of SYMPTOMs that can be accounted for by a set of EXPLANATORY_DISORDER. Formally, it is defined as,

$$intellego : coverage(e_1, e_2, ..., e_n) \equiv \exists \texttt{IS\_SYMPTOM\_OF}\{e_1\} \sqcup \exists \texttt{IS\_SYMPTOM\_OF}\{e_2\} \sqcup \cdots \sqcup \exists \texttt{IS\_SYMPTOM\_OF}\{e_n\},$$

where $e_i, i = 1, 2, 3, ..., n$ are instances of EXPLANATORY_DISORDER.

The following example demonstrates the population of instances of *'intellego:coverage'* class for a document which reports *atrial fibrillation* and *hypertension* as disorders.

$$intellego : coverage(hypertension, atrialfibrillation) \equiv \exists \texttt{IS\_SYMPTOM\_OF}\{hypertension\} \sqcup \exists \texttt{IS\_SYMPTOM\_OF}\{atrialfibrillation\}$$

An OWL reasoner is used to populate instances of *'intellego:coverage'* class. The *'intellego:coverage'* class will be referred to as COVERED_SYMPTOM from here onwards. The clinical narrative is *consistent* if all the instances of OBSERVED_SYMPTOM are a subset of COVERED_SYMPTOM.

$$isConsistent \quad iff \quad \texttt{OBSERVED\_SYMPTOM} \subseteq \texttt{COVERED\_SYMPTOM} \tag{5.1}$$

All symptoms that are in OBSERVED_SYMPTOM but not in COVERED_SYMPTOM are identified as unexplained symptoms and one of the reasons for unexplained symptoms is the incompleteness of the knowledge in entity models. The information extraction step of the solution developed in this chapter focuses on ac-

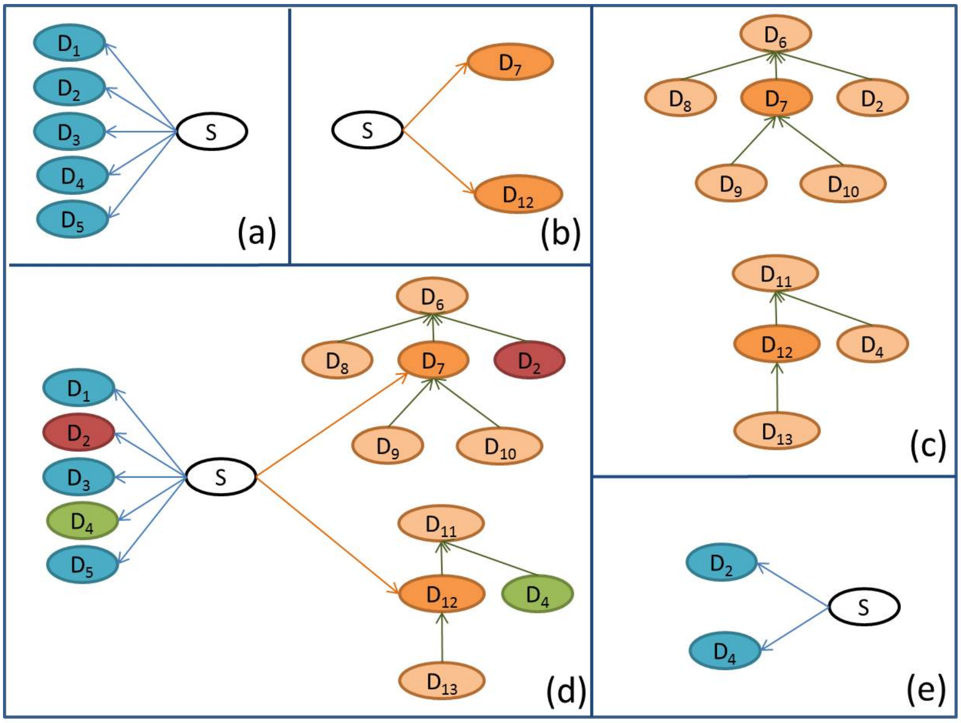


Figure 5.5: Suggest Candidate Relationships

quiring this missing knowledge and ultimately extracts such implicit relationships between entities in clinical narratives.

5.3.5 Implicit Information Extraction - Identifying Unknown Non-hierarchical Relationships

With the consistency assumption on the clinical narratives, all disorders in EXPLANATORY_DISORDER become candidates for accounting for unexplained symptoms. However, it is a tedious task to investigate potential IS_SYMPTOM_OF relationships between unexplained symptoms and all disorders in EXPLANATORY _DISORDER. Hence, the information extraction step uses knowledge about the disorders in EXPLANATORY _DISORDER and unexplained symptom to generate hypotheses that have the highest potential to be an explanation for the unexplained symptoms. The following steps describe how the most plausible relationships are filtered out.

1. Collect a set of disorders from entity models which are related to the unexplained symptom. This set is referred to as 'known disorders'.

2. Collect disorders that appear in the 'neighborhood' of each known disorder. The hierarchical relationships are used to find the UMLS entities in the neighborhood. Specifically, parents, children, and siblings of a disorder are included in the neighborhood.

3. Perform union operation on collected neighborhoods and intersect that with the set of disorders in `EXPLANATORY_DISORDER` to obtain the filtered candidate disorder set.

Figure 5.5 depicts the steps in this process. Let symptom S be an unexplained symptom in the document and disorders D_1, D_2, D_3, D_4, D_5 be present in the same document (Figure 5.5(a)) (i.e., they form an `EXPLANATORY_DISORDER` set). Initially, all these disorders are members of the candidate disorder set. From the modeled knowledge, it is found that symptom S is a symptom of two other disorders, namely D_7 and D_{12} (Figure 5.5(b)). With this extra knowledge, it collects the neighborhoods for D_7 and D_{12} as depicted in Figure 5.5(c). It turns out that D_2 and D_4 are members of both the initial candidate set and the collected neighborhoods (Figure 5.5(d)). This suggests that D_2 and D_4 are the most probable candidates that can explain symptom S, which results in eliminating D_1, D_3, and D_5 from the initial candidate set (Figure 5.5(e)).

The intuition behind this technique is that a symptom is most likely to be shared by similar disorders. The similar disorders are collected by exploiting the UMLS entity hierarchy.

Selected candidate relationships are presented to the domain expert in the form of questions. The answers to those questions reveal the implicit `IS_SYMPTOM_OF` relationships that were not possible to identify with the given knowledge base. The outcome of this step can also be used to update the knowledge base with more non-hierarchical relationships. i.e., if the experts agree with the suggested relationship between the disorder and the symptom, it is added to the background knowledge, otherwise it is ignored. Since this step adds more knowledge to the knowledge base, this step is run as many times as necessary to find more relationships. The algorithm terminates when there are no new questions to be generated. Since there are only a finite number of elements in the candidate set, the termination of the algorithm is guaranteed. Although this dissertation limits the implementation of this algorithm to find the relationships between disorders and symptoms, the idea can be applied to find the relationships between medications and symptoms/disorders, and procedures and disorders.

5.4 Evaluation

The solution was implemented in Java, and the OWL API[3] was used to interact with the ontology. The Pellet reasoner[4] was used for the reasoning task of finding the instances of class COVERED_SYMPTOM. The evaluation was conducted on 1,500 unstructured de-identified EMRs.

5.4.1 Building Initial Knowledge Base

To build the initial knowledge base, entities in the corpus of 1,500 EMRs were linked with cTAKES and the most frequent entities were selected. The semantic types in UMLS were used to categorize frequent entities into symptoms and disorders. The entities belonging to the semantic types “Finding (T033)” and “Sign or Symptom (T184)” were categorized as symptoms and entities belonging to “Disease and Syndrome (T047)” were categorized as disorders. There were few entities that do not belong to these categories. The domain experts were consulted to categorize the semantic types of such concepts into disorder or symptom. For example, *atrial fibrillation* belongs to “Pathologic Function (T046)” in UMLS and is categorized as disorder by the domain expert. The initial knowledge base consisted of 86 disorders and 42 symptoms. The next task was to identify an initial set of IS_SYMPTOM_OF relationships. 255 IS_SYMPTOM_OF relationships were identified using the Web resources and by consulting domain experts.

[3] http://owlapi.sourceforge.net

[4] http://clarkparsia.com/pellet

5.4.2 Evaluating Unknown Implicit Relationship Extraction

The implicit relationship extraction algorithm is executed on each clinical document and corpus-wide statistics were collected. Specifically, the solution calculates how many times a particular symptom was found as unexplained and what disorders were present in the `EXPLANATORY_DISORDER` set in such instances and their frequency. Whenever symptom S is found as unexplained, it is called an *unexplained instance* of symptom S. There were 29 unexplained symptoms in the evaluation corpus. Table 5.1 contains the top 10 unexplained symptoms. Table 5.2 contains most frequent disorders found in the `EXPLANATORY_DISORDER` set for the entity *edema* when *edema* was found to be unexplained. According to Table 5.1, *edema* is found to be unexplained in 910 documents. The two rows of Table 5.2 say that *hypertension* and *hyperlipidemia* were present 647 times and 641 times within those 910 instances respectively.

Once the unexplained instances of symptoms and disorders in `EXPLANATORY_DISORDER` in each instance are identified, the next task is to suggest the relationships that can rectify the unexplained instances. This task is evaluated with precision and recall metrics.

5.4.3 Precision of Suggested Relationships

The solution makes effective and efficient use of the domain experts' availability for unknown implicit relationship extraction. Hence the precision of the generated questions is an important evaluation metric. The *precision* is defined as:

$$precision = \frac{\textit{number of correct relationships suggested}}{\textit{total number of relationships suggested}} * 100$$

Table 5.3 summarizes the precision of suggested relationships. The suggested relationships in the first round has a precision of 73.94% and 105 new relationships were added to the initial knowledge base. Iteration 2 was run with new knowledge and another 20 correct relationships were suggested with 68.96% precision.

It was decided to terminate the algorithm after the 2nd iteration since the 3rd iteration added only 4 relationships with poor precision. The overall experiment suggested 171 relationships out of which 125 were correct, yielding a precision of 73.09%.

Symptom	number of unexplained instances
edema	910
syncope	336
systolic murmur	168
tachycardia	143
angina	136
depression	114
dyspnea	64
hypotension	58
headache	45
fatigue	30

Table 5.1: Top 10 unexplained symptoms

Disorder	number of co-occurrences
hypertension	647
hyperlipidemia	641
claudication	454
coronary atherosclerosis	395
coronary artery disease	242
cardiac arrhythmia	232
diabetes mellitus	213
arthritis	176
apnea	165
atrial fibrillation	158

Table 5.2: Statistics of disorders in `EXPLANATORY_DISORDER` with unexplained instances of entity *edema*

Iteration	number of suggestions	number of correct	*precision*
1	142	105	73.94%
2	29	20	68.96%
3	9	4	44.44%

Table 5.3: The Precision of suggested relationships

Symptom	# co-occurring disorders (COD)	# correct in COD	# incorrect in COD	# correct suggestions	# incorrect suggestions
chest pain	3	0	3	0	0
numbness	8	3	5	1	1
nausea	5	1	4	1	1
dyspnea	6	1	5	0	2
angina	11	5	6	4	1
asthenia	1	0	1	0	0
tachypnea	1	0	1	0	0
coughing	2	0	2	0	2
wheezing	6	2	4	1	2
chest tightness	8	5	3	4	0
systolic murmurs	13	1	12	0	0
chest pain(left side)	2	1	1	1	0
swelling	9	4	5	0	0
paralysis	9	2	7	0	0
abdominal pain	4	3	1	2	0
dizziness	4	3	1	2	0

fatigue	5	2	3	2	1
hypokalemia	4	0	4	0	0
soreness	8	2	6	0	0
headache	8	3	5	2	1
hypotension	11	1	10	0	0
depression	12	4	8	0	0
edema	13	5	8	3	1
syncope	11	2	9	2	3
rhonchi	8	0	8	0	0
tachycardia	11	6	5	5	0
discomfort in chest	6	1	5	1	0
cyanosis	2	1	1	0	0
hyperkalemia	9	0	9	0	0
Total	**200**	**58**	**142**	**31**	**15**

Table 5.4: Comparison of the output with a simpler method

A much simpler method for finding unknown implicit relationships is to suggest all disorders in EXPLANATORY_DI when a symptom is found to be unexplained (co-occurrence based method). The strength of the developed knowledge-based method over this simpler method is its ability to filter out disorders which are unlikely to have a relationship with the symptoms, even though the disorders co-occur with an unexplained symptom. The next evaluation compares the results obtained by the simpler method and the knowledge-based method developed in this chapter. As mentioned before, there were 29 unexplained symptoms in the corpus. Each of these unexplained symptoms co-occur with more than one disorder. The 29 unexplained symptoms have a total of 947 such co-occurrences. But due to the limited availability of the domain experts, it is not possible to validate all these co-occurrences. Hence, it was decided to validate the top co-occurring disorders (based on co-occurrence frequency) of each symptom and compare the results with knowledge-based method. Specifically, the top 20% of the co-occurring disorders of each symptom were collected and domain experts were asked to mark the correct relationships among them and compare these results with the knowledge-based method. Table 5.4 show the results of this experiment. According to Table 5.4, the top 20% of co-occurring disorders with the unexplained symptom *angina* consist of 11 disorders. Within these 11 disorders, 5 of them have relationships with *angina*. the knowledge-based method suggested 4 out of 5, and suggested only one incorrect relationship. The symptom *rhonchi* had 8 disorders in top 20% of co-occurring disorders. However, none of them have relationship with *rhonchi*. The knowledge-based method able to figure out this irrelevancy between *rhonchi* and top co-occurring disorders and did not suggest any of them while co-occurring based method would suggest all of them as candidate relationship leading to 0% precision.

In summary, there were a total of 200 relationships within the top 20% of co-occurring disorders for each unexplained symptom, out of which 58 were correct. Hence, the co-occurrence based method would have had a precision of 29.0% (58/200). The knowledge-based method suggested 31 out of 58 correct relationships while suggesting 15 incorrect relationships with a precision of 67.39% (31/46) and a recall of 53.44% (31/58). This experiment shows that existing knowledge of the entities was able to filter out incorrect relationships to improve the precision significantly when compared to the simpler method while also maintaining good recall.

5.4.4 Recall of Suggested Relationships

The knowledge-based approach is capable of finding relationships between symptoms and disorders that are missing in the given knowledge base but present in real clinical documents. Due to the limited availability of the domain experts it was not possible to conduct an experiment to calculate the recall for 1,500 EMRs. Instead, this evaluation randomly selected 30 clinical documents and asked domain experts to find all the relationships between disorders and symptoms that existed in these documents. For instance, if a clinical document contains 3 disorders and 4 symptoms, there are 12 possible relationships. The domain experts were asked to select the correct relationships among these 12 relationships. Let's assume the domain experts found 7 relationships, 3 of which were already present in the given knowledge base. Then, the task of

All Correct Relationships	109
Known Correct Relationships	42
Found Correct Relationships	30
Not Found Correct Relationships	37
Recall	45.45%

Table 5.5: Recall of suggested relationships for 30 clinical documents

knowledge-based method is to find the remaining 4 relationships. Hence the recall is defined as follows and Table 5.5 shows the results of this experiment.

$$recall = \frac{\textit{correct relationships found}}{\textit{all correct relationships - known correct relationships}} * 100$$

‘correct relationship found’ is the number of correct relationships found by knowledge-based method, ‘all correct relationships’ is the number of all correct relationships among symptoms and disorders, and ‘known correct relationships’ is the number of already known relationships among ’all correct relationships’. In other words, the denominator is the number of unknown correct relationships that exist, while the numerator is what is found by the algorithm.

There were two main reasons for the low recall in Table 5.5.

- If at least one disorder explains the symptom in the document then it is not found as an unexplained symptom:

  A symptom is not identified as unexplained if there is at least one disorder in the document that can explain the symptom and the knowledge base has this relationship. This prevents the suggestion of other co-occurring disorders within the document as candidates for having relationship with the symptom even if they are related to this symptom. For example, consider a clinical document document with *edema*, *congestive heart failure*, *hypertension*, and *cardiomyopathies*. The knowledge base has a relationship between the symptom *edema* and the disorder *hypertension*. This makes *edema* explainable within this document; hence the knowledge-based approach does not suggest the other two disorders as candidates

All Correct Relationships	109
Known Correct Relationships	43
Found Correct Relationships	44
Not Found Correct Relationships	22
Recall	66.67%

Table 5.6: Recall for the relationships found in 30 EMRs when executed with more data

that have a relationship with *edema* although they actually do have such a relationship. Therefore, these two relationships were missed from this clinical document. However, given more data that do not contain these comorbidities (concurrent disorders), the knowledge-based approach is capable of finding these relationships. For example, if there is a clinical document within the given corpus which has *edema*, *congestive heart failure*, and *cardiomyopathies* but not *hypertension*, *edema* becomes unexplained and both *congestive heart failure* and *cardiomyopathies* become plausible candidates for a relationship with *edema*, and these two relationships can be suggested and eventually validated.

- The neighborhood method cannot reach the disorder:

  Even though the disorder *cardiomyopathies* co-occurs with the symptom *edema* in the above scenario, if none of the neighbors of *cardiomyopathies* have a relationship to *edema* in the current knowledge base, it is not collected in neighborhood collection step. Hence, it will never suggest this relationship.

The experiment with 30 EMR documents missed 37 correct relationships as shown in Table 5.5. The lack of different combinations of comorbidities within 30 documents significantly contributes towards this result. Hence, given more data, this method should be able to find these relationships. To demonstrate this, 400 documents were selected from corpus and the algorithm was executed. The intention was to check how many of the missed relationships were suggested with more data. Table 5.6 contains the improved recall for relationships found in the 30 documents above given more data.

Knowledge base	**number of unexplained instances**	**increment in explanability**
with initial knowledge base	2251	0%
after iteration 1	878	60.99%
after iteration 2	806	64.19%

Table 5.7: Comparison of explainability before and after running the knowledge-based method

5.4.5 Increment of Explanatory Power of the Knowledge Base

The overall goal of the algorithm is to extract unknown implicit relationships that are not present in the given knowledge base. Hence it is necessary to quantify the impact of new relationships that were added to the knowledge base. This is done by quantifying the difference of explainability of the clinical documents before and after extracting unknown implicit relationships. We define the explanatory power as,

$$E = \textit{\# of explainable instances in data set}, \tag{5.2}$$

where E is explainability and the increment of explainability is defined as:

$$\textit{increment of EP} = \frac{UI_i - UI_n}{UI_i} * 100, \tag{5.3}$$

where UI_i is the number of unexplained instances of a given data set before extracting implicit relationships with the knowledge-based method and UI_n is the number of unexplained instances after running the knowledge-based method.

As Table 5.7 shows (UI stands for the number of unexplained instances), the explainability is increased by 39.87% at the end of the second iteration.

The implemented algorithm depends on the output of the entity linking tool. The above experiments were conducted on the clinical corpus processed with the cTAKES. The limitations of the cTAKES output directly

Symptom	number of unexplained instances
edema	206
depression	172
angina	134
dyspnea	120
syncope	103
tachycardia	88
chest discomfort	72
headache	68
chest pain	59
fatigue	52

Table 5.8: Top 10 unexplained symptoms in the corpus processed with MedLEE

impact the unknown implicit relationship extraction algorithm. For example, if cTAKES does not identify an entity present in the clinical narrative, that would have explained a symptom according to the knowledge base, the algorithm identifies this symptom as unexplained and would suggest plausible relationships. These relationships are most likely to turn down by the domain expert. In order to compare the results of the algorithm, the above experiments were repeated with a corpus processed with MedLEE clinical text processing tool [Friedman et al. 1994].

Table 5.8 shows the top 10 unexplained symptoms in the corpus parsed with MedLEE and Table 5.9 shows the top 10 co-occurring disorders with the unexplained instances of *edema*.

The next experiment is conducted to measure the accuracy of the relationships suggested by using the corpus parsed with MedLEE. Table 5.10 shows the link accuracy of the predicted relationships using this corpus.

The comparison of the relationships generated with MedLEE output and the simpler method is shown in Table 5.11. As shown, the number of relationships suggested by the corpus processed with MedLEE is

Disorder	number of co-occurrences
hyperlipidemia	116
hypertension	112
atrial fibrillation	84
coronary artery disease	66
coronary arteriosclerosis	62
diabetes mellitus	45
hypothyroidism	41
GERD	34
COPD	25
PVD	24

Table 5.9: Statistics of disorders in `EXPLANATORY_DISORDER` with unexplained instances of entity *edema* in the corpus parsed with MedLEE

Iteration	number of suggestions	number of correct	*precision*
1	98	76	77.55%
2	16	10	62.50%
3	8	4	50%

Table 5.10: The Precision of suggested relationships with MedLEE corpus

lower than the corpus processed with cTAKES. This is due to the fact that MedLEE found less number of unexplained symptoms and instances as shown in Table 5.8.

5.5 Limitations

As shown in the evaluation, the knowledge-based solution has good precision in suggesting relationships, but it has the following limitations.

- It is unable to deal with complex relationships.

  The knowledge base contains only single symptom to single disorder relationships, but it is possible that a single symptom can be explained by the existence of multiple disorders. This method is not able to capture such complex relationships.

- It may still miss potential relationships.

  The algorithm might miss some relationships in EMR document due to two reasons: 1) If the same symptom can be explained by multiple disorders in a clinical document, it may not attribute the symptom to all of them. 2) If none of the neighbors of a co-occurring disorder has a relationship with the unexplained symptom, it may miss considering them as candidates for suggesting a relationship.

- The precision of the suggested relationships depends on the precision of the NLP engine.

  The proposed method requires the NLP engine to annotate the entities and associate negation and temporal information with the entity. The errors in the NLP output can affect the precision of proposed method.

5.6 Conclusion

The relationships in clinical narratives are rarely expressed explicitly. It is necessary to figure out the relationships between entities in clinical documents to understand the content of the document, and consequently, the patient's health status. The common strategy uses relationships stated in a knowledge base to popu-

Symptom	# co-occurring disorders (COD)	# correct in COD	# incorrect in COD	# correct suggestions	# incorrect suggestions
chest pain	3	0	3	0	0
nausea	3	1	2	1	0
fatigue	3	2	1	2	1
hypokalemia	4	0	4	0	0
tachypnea	1	0	1	0	0
headache	5	2	3	2	0
distress	4	1	3	1	0
hypotension	6	1	5	0	0
coughing	3	0	3	0	2
angina	7	5	2	4	0
wheezing	4	0	4	0	0
depression	7	2	5	0	0
numbness	6	4	2	2	0
paralysis	5	2	3	0	0
dyspnea	3	0	3	0	1
edema	7	5	2	3	0
syncope	6	2	4	2	3
abdominal pain	3	3	0	3	0
rhonchi	5	0	5	0	0
dizziness	3	1	2	1	0
hypotension	11	1	10	0	0
discomfort in chest	5	2	3	1	0
tachycardia	5	4	1	3	0
hyperkalemia	5	0	5	0	0
Total	**103**	**37**	**66**	**25**	**7**

late relationships between entities in clinical documents. However, it has been found that knowledge bases with clinical knowledge are incomplete with respect to relationships and acquiring such knowledge by using domain experts is a tedious task that can result in collecting subjective knowledge. This chapter took the challenge of filling in the gaps in existing clinical knowledge bases and, ultimately, populating the unknown implicit relationships in clinical documents.

The developed solution uses a human in the loop model to acquire the knowledge. The challenge is to minimize the human effort and time since optimal use of domain experts is necessary. The approach uses the hierarchical knowledge and known non-hierarchical relationships between entities to improve the accuracy of the generated hypotheses. The hypotheses are validated by the domain experts and the evaluation showed the effectiveness of the developed knowledge-based method when compared to a pure statistical method in acquiring the required knowledge. Ultimately this approach improved the ability to extract implicit relationships in clinical documents.

6

Conclusion and Future Work

This chapter summarizes the findings of this dissertation and talk about interesting future research directions.

6.1 Summary

People can communicate their ideas, opinions, and facts in an implicit manner. It is found that implicit constructs in language carry a unique value in day to day communication. Failure to identify the implicit information in the text can adversely effect the downstream applications such as sentiment analysis, trend detection, computer assisted coding in clinical documents, and secondary analysis applications on clinical data (e.g., prediction tasks).

This dissertation addressed the problem of extracting implicit information from the text. It focused on extracting implicit entities and relationships in the text. For example, it looked at how to identify the implicit mention of the movie *Gravity* in the tweet *'New Sandra Bullock astronaut lost in space movie looks absolutely terrifying'* and how to identify the relationship between drug *Dolasteron* and clinical condition *nausea* in the text *'He is suffering from nausea and severe headaches. Dolasteron was prescribed.'*

It is observed that the shared understanding/knowledge about the topic being discussed between the speaker and the audience allows to decode the implicit information in daily communication. In order to simulate this process with an algorithm, it is required to have this shared knowledge in machine readable

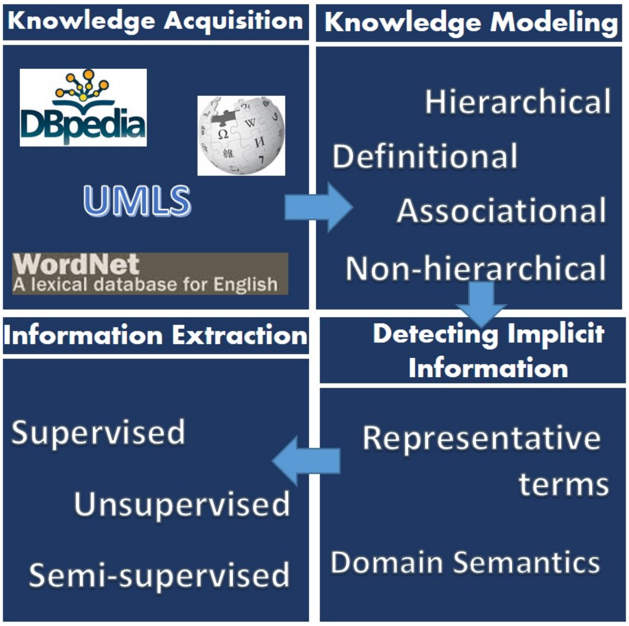


Figure 6.1: Summary of the developed framework

manner. Hence, one of the main components of the solution developed in this dissertation is the background knowledge about the domain. The solution starts by identifying the sources with the relevant background knowledge, and it acquires, processes and models the background knowledge. These models provide the necessary infrastructure to perform implicit information extraction. The next step of the solution identifies the text with implicit information and extracts them. Figure 6.1 shows main components of the solution.

The developed solution used DBpedia, Wikipedia, and UMLS to acquire relevant domain knowledge for each task and the WordNet to acquire linguistic knowledge. It creates the knowledge models using hierarchical, non-hierarchical, definitional, and associational knowledge about the entities. As the last step, it deploys supervised, semi-supervised, and unsupervised solution to extract the implicit information.

The dissertation demonstrates the effectiveness of the developed solution by applying it to link implicit entities in clinical narratives and tweets, and by extracting implicit relationships between entities in clinical narratives.

6.2 Future Work

This is one of the early attempts to extract implicit entities and relationships from the text. The following tasks are identified as interesting directions for future research.

- Incorporate more knowledge sources: The current work can be benefitted by exploiting more knowledge sources. For example, although DBpedia is used widely as a background knowledge base for various tasks, it lacks some factual knowledge on entities. For instance, it does not have a single *'starred in'* relationship for the movie *Guardian of Galaxy*. However, such knowledge is available in other knowledge bases in linked open data cloud which can be used as complementary knowledge sources. Wikipedia is a another rich knowledge base which is being updated as a collective effort by the crowd. The knowledge in Wikipedia is not readily available in structured format. However, one can exploit its hyperlink and text structure to extract knowledge that can complement the knowledge sources used in this dissertation. The knowledge extracted with more knowledge sources can be used to model richer representations that can ultimately improve the accuracy and coverage of implicit information extraction task.

- Capture temporal dynamics of the domain knowledge: The entity types considered in tweets are associated with highly dynamic knowledge. In fact, one of the reasons for failures in linking implicit entities in tweets is the lack of up to date knowledge about the entities. Hence, it is important to be able to update the modelled domain knowledge in response to real world events. There are two events that can affect the modelled knowledge over time: 1) A new entity becomes popular and people start to talk about it or the popularity of an existing entity faded away and people no longer talk about it, 2) A new topic of the interest emerged for an existing entity or with the introduction of a new entity or the popularity of the existing topic faded away. Hence, it would be a interesting to develop operators that can account for these dynamics in the real world and keep the created models updated.

- Improve the techniques developed to detect the presence of implicit information in text: The developed framework has a component that is responsible to identify the text with implicit information. The limitations shown by the applications demonstrated in this dissertation are due to the hindrances of the techniques deployed within this step of the solution. For example, in implicit entity linking scenario, the text (tweet or clinical text) does not become a candidate to have an implicit entity mention unless it contains one of the selected semantic cues; in implicit relationship extraction scenario, if the symptom has relationship with at least one of the disorders in the clinical note it wont be detected as a unexplained symptom, consequently, would not consider in the step that extracts unknown relationships. These limitations can be overcome by developing a more sophisticated solution to identify the text with implicit information by analyzing semantic and syntactic features of the text.

7

Bibliography

ABULAISH, M. AND DEY, L. 2007. Biological relation extraction and query answering from medline abstracts using ontology-based text mining. *Data & Knowledge Engineering 61,* 2, 228–262.

ACLWIKI. 2014. Paraphrase identification (state of the art). `http://aclweb.org/aclwiki/index.php?title=Paraphrase_Identification_(State_of_the_art)`. [Online; accessed 19-Dec-2014].

AONE, C. AND RAMOS-SANTACRUZ, M. 2000. Rees: a large-scale relation and event extraction system. In *Proceedings of the sixth conference on Applied natural language processing*. Association for Computational Linguistics, 76–83.

ARONSON, A. R. 2006. Metamap: Mapping text to the umls metathesaurus. *Bethesda, MD: NLM, NIH, DHHS*.

ASHBURNER, M., BALL, C. A., BLAKE, J. A., BOTSTEIN, D., BUTLER, H., CHERRY, J. M., DAVIS, A. P., DOLINSKI, K., DWIGHT, S. S., EPPIG, J. T., ET AL. 2000. Gene ontology: tool for the unification of biology. *Nature genetics 25,* 1, 25–29.

BANERJEE, S. AND PEDERSEN, T. 2002. An adapted lesk algorithm for word sense disambiguation using wordnet. In *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer, 136–145.

BARZILAY, R. AND ELHADAD, N. 2003. Sentence alignment for monolingual comparable corpora. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*. Association for Computational Linguistics, 25–32.

BLANCO, E., CASTELL, N., AND MOLDOVAN, D. I. 2008. Causal relation extraction. In *LREC*.

BODENREIDER, O. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research 32,* suppl 1, D267–D270.

BRAGG, M. 2006. *The adventure of English: The biography of a language*. Arcade Publishing.

BUNESCU, R. C. AND PASCA, M. 2006. Using encyclopedic knowledge for named entity disambiguation. In *EACL*. Vol. 6. 9–16.

CHANDRASEKARAN, B., JOSEPHSON, J. R., AND BENJAMINS, V. R. 1999. What are ontologies, and why do we need them? *IEEE Intelligent systems* 1, 20–26.

CHANG, M.-W., HSU, B.-J., MA, H., LOYND, R., AND WANG, K. 2014. E2e: An end-to-end entity linking system for short and noisy text. *Making Sense of Microposts*.

CHAPMAN, W. W., BRIDEWELL, W., HANBURY, P., COOPER, G. F., AND BUCHANAN, B. G. 2001. A simple algorithm for identifying negated findings and diseases in discharge summaries. *Journal of biomedical informatics 34,* 5, 301–310.

CHOMSKY, N. 1964. Aspects of the theory of syntax. Tech. rep., DTIC Document.

CHUN, H.-W., TSURUOKA, Y., KIM, J.-D., SHIBA, R., NAGATA, N., HISHIKI, T., AND TSUJII, J. 2006. Extraction of gene-disease relations from medline using domain dictionaries and machine learning. In *Pacific Symposium on Biocomputing*. Vol. 11. 4–15.

CIARAMITA, M., GANGEMI, A., RATSCH, E., SARIC, J., AND ROJAS, I. 2005. Unsupervised learning of semantic relations between concepts of a molecular biology ontology. In *IJCAI*. 659–664.

COLLINS, M. AND SINGER, Y. 1999. Unsupervised models for named entity classification. In *Proceedings of the joint SIGDAT conference on empirical methods in natural language processing and very large corpora*. Citeseer, 100–110.

CORTES, C. AND VAPNIK, V. 1995. Support-vector networks. *Machine learning 20,* 3, 273–297.

CRUSE, D. A. 1986. *Lexical semantics*. Cambridge University Press.

CUCERZAN, S. 2007. Large-scale named entity disambiguation based on wikipedia data. In *EMNLP-CoNLL*. Vol. 7. 708–716.

DAIBER, J., JAKOB, M., HOKAMP, C., AND MENDES, P. N. 2013. Improving Efficiency and Accuracy in Multilingual Entity Extraction. In *Proceedings of the 9th International Conference on Semantic Systems (I-Semantics)*.

DAVIDOV, D., TSUR, O., AND RAPPOPORT, A. 2010. Semi-supervised recognition of sarcastic sentences in twitter and amazon. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics, 107–116.

DERCZYNSKI, L., MAYNARD, D., RIZZO, G., VAN ERP, M., GORRELL, G., TRONCY, R., PETRAK, J., AND BONTCHEVA, K. 2015. Analysis of named entity recognition and linking for tweets. *Information Processing & Management 51,* 2, 32–49.

DO, Q. X., CHAN, Y. S., AND ROTH, D. 2011. Minimally supervised event causality identification. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 294–303.

DOLAN, B., QUIRK, C., AND BROCKETT, C. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th international conference on Computational Linguistics*. Association for Computational Linguistics, 350.

DOWTY, D. 1991. Thematic proto-roles and argument selection. *Language*, 547–619.

DREDZE, M., MCNAMEE, P., RAO, D., GERBER, A., AND FININ, T. 2010. Entity disambiguation for knowledge base population. In *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics, 277–285.

ENGLISH LANGUAGE. 2001. English language — Wikipedia, the free encyclopedia. [Online; accessed 02-11-2016].

FAURE, D. AND NÉDELLEC, C. 1998. A corpus-based conceptual clustering method for verb frames and ontology acquisition. In *LREC workshop on adapting lexical and corpus resources to sublanguages and applications*. Vol. 707. 30.

FELLBAUM, C. 1998. *WordNet*. Wiley Online Library.

FERNANDO, S. AND STEVENSON, M. 2008. A semantic similarity approach to paraphrase detection. In *Proceedings of the 11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics*. Citeseer, 45–52.

FERRAGINA, P. AND SCAIELLA, U. 2010. Tagme: on-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM international conference on Information and knowledge management*. ACM, 1625–1628.

FERRUCCI, D., BROWN, E., CHU-CARROLL, J., FAN, J., GONDEK, D., KALYANPUR, A. A., LALLY, A., MURDOCK, J. W., NYBERG, E., PRAGER, J., ET AL. 2010. Building watson: An overview of the deepqa project. *AI magazine 31,* 3, 59–79.

FILLMORE, C. J. 1976. Frame semantics and the nature of language*. *Annals of the New York Academy of Sciences 280,* 1, 20–32.

FRIEDMAN, C., ALDERSON, P. O., AUSTIN, J. H., CIMINO, J. J., AND JOHNSON, S. B. 1994. A general natural-language text processor for clinical radiology. *Journal of the American Medical Informatics Association 1,* 2, 161–174.

FRIEDMAN, C., SHAGINA, L., LUSSIER, Y., AND HRIPCSAK, G. 2004. Automated encoding of clinical documents based on natural language processing. *Journal of the American Medical Informatics Association 11,* 5, 392–402.

FU, X. AND ANANIADOU, S. 2014. Improving the extraction of clinical concepts from clinical records. *Proceedings of BioTxtM14*.

GIAMPICCOLO, D., MAGNINI, B., DAGAN, I., AND DOLAN, B. 2007. The third pascal recognizing textual entailment challenge. In *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*. Association for Computational Linguistics, 1–9.

GIRJU, R., MOLDOVAN, D. I., ET AL. 2002. Text mining for causal relations. In *FLAIRS Conference*. 360–364.

GRUBER, T. R. 1993. A translation approach to portable ontology specifications. *Knowledge acquisition 5,* 2, 199–220.

GUO, S., CHANG, M.-W., AND KICIMAN, E. 2013. To link or not to link? a study on end-to-end tweet entity linking. In *HLT-NAACL*. 1020–1030.

HACHEY, B., RADFORD, W., NOTHMAN, J., HONNIBAL, M., AND CURRAN, J. R. 2013. Evaluating entity linking with wikipedia. *Artif. Intell. 194*, 130–150.

HEARST, M. A. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of the 14th conference on Computational linguistics-Volume 2*. Association for Computational Linguistics, 539–545.

HENSON, C., THIRUNARAYAN, K., AND SHETH, A. 2011. An ontological approach to focusing attention and enhancing machine perception on the web. *Applied Ontology 6,* 4, 345–376.

HOFFART, J., ALTUN, Y., AND WEIKUM, G. 2014. Discovering emerging entities with ambiguous names. In *Proceedings of the 23rd international conference on World wide web*. International World Wide Web Conferences Steering Committee, 385–396.

HOFFART, J., YOSEF, M. A., BORDINO, I., FÜRSTENAU, H., PINKAL, M., SPANIOL, M., TANEVA, B., THATER, S., AND WEIKUM, G. 2011. Robust disambiguation of named entities in text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 782–792.

HSU, M.-H., TSAI, M.-F., AND CHEN, H.-H. 2008. Combining wordnet and conceptnet for automatic query expansion: a learning approach. In *Asia information retrieval symposium*. Springer, 213–224.

JACKENDOFF, R. 1987. The status of thematic relations in linguistic theory. *Linguistic inquiry 18,* 3, 369–411.

JIANG, J. J. AND CONRATH, D. W. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. *arXiv preprint cmp-lg/9709008*.

KATZ, J. J. AND FODOR, J. A. 1963. The structure of a semantic theory. *language 39,* 2, 170–210.

KAVALEC, M., MAEDCHE, A., AND SVÁTEK, V. 2004. Discovery of lexical entries for non-taxonomic relations in ontology learning. In *International Conference on Current Trends in Theory and Practice of Computer Science*. Springer, 249–256.

KHOO, C., CHAN, S., NIU, Y., AND ANG, A. 1999. A method for extracting causal knowledge from textual databases. *Singapore journal of library & information management 28*, 48–63.

KIPPER, K., DANG, H. T., PALMER, M., ET AL. 2000. Class-based construction of a verb lexicon. In *AAAI/IAAI*. 691–696.

KOLOWICH, L. 2016. The handy character count guide for blog posts, facebook pages and more. [Online; accessed 06-29-2016].

LEACOCK, C. AND CHODOROW, M. 1998. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database 49,* 2, 265–283.

LEE, K. 2014. Infographic: The optimal length for every social media update and more. [Online; accessed 06-29-2016].

LEHMANN, J., ISELE, R., JAKOB, M., JENTZSCH, A., KONTOKOSTAS, D., MENDES, P. N., HELLMANN, S., MORSEY, M., VAN KLEEF, P., AUER, S., ET AL. 2014. Dbpedia-a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web Journal 5*, 1–29.

LI, X., SZPAKOWICZ, S., AND MATWIN, S. 1995. A wordnet-based algorithm for word sense disambiguation. In *IJCAI*. Vol. 95. 1368–1374.

LIN, D. 1998. An information-theoretic definition of similarity. In *ICML*. Vol. 98. 296–304.

LIU, H. AND SINGH, P. 2004. Conceptneta practical commonsense reasoning tool-kit. *BT technology journal 22,* 4, 211–226.

LIU, X., LI, Y., WU, H., ZHOU, M., WEI, F., AND LU, Y. 2013. Entity linking for tweets. In *ACL (1)*. 1304–1311.

MARTHA, P., DAN, G., AND PAUL, K. 2005. The proposition bank: a corpus annotated with semantic roles. *Computational Linguistics Journal 31*, 1.

MCCAWLEY, J. D. 1976. *Grammar and meaning: papers on syntactic and semantic topics*. Academic Press.

MEIJ, E., WEERKAMP, W., AND DE RIJKE, M. 2012. Adding semantics to microblog posts. In *Proceedings of the fifth ACM international conference on Web search and data mining*. ACM, 563–572.

MENDES, P. N., JAKOB, M., GARCÍA-SILVA, A., AND BIZER, C. 2011. Dbpedia spotlight: shedding light on the web of documents. In *Proceedings of the 7th international conference on semantic systems*. ACM, 1–8.

MIHALCEA, R., CORLEY, C., AND STRAPPARAVA, C. 2006. Corpus-based and knowledge-based measures of text semantic similarity. In *AAAI*. Vol. 6. 775–780.

MIKOLOV, T., CHEN, K., CORRADO, G., AND DEAN, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

MILLER, G. AND FELLBAUM, C. 1998. Wordnet: An electronic lexical database.

MILLER, G. A., BECKWITH, R., FELLBAUM, C., GROSS, D., AND MILLER, K. J. 1990. Introduction to wordnet: An on-line lexical database*. *International journal of lexicography 3,* 4, 235–244.

MILLER, G. A. AND FELLBAUM, C. 1991. Semantic networks of english. *Cognition 41,* 1, 197–229.

MILNE, D. AND WITTEN, I. H. 2008. Learning to link with wikipedia. In *Proceedings of the 17th ACM conference on Information and knowledge management*. ACM, 509–518.

MINTZ, M., BILLS, S., SNOW, R., AND JURAFSKY, D. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2-Volume 2*. Association for Computational Linguistics, 1003–1011.

MOONEY, R. J. AND BUNESCU, R. C. 2005. Subsequence kernels for relation extraction. In *Advances in neural information processing systems*. 171–178.

MULKAR-MEHTA, R., WELTY, C., HOOBS, J. R., AND HOVY, E. 2011. Using granularity concepts for discovering causal relations. In *Proceedings of the FLAIRS conference*.

NADEAU, D. AND SEKINE, S. 2007. A survey of named entity recognition and classification. *Lingvisticae Investigationes 30,* 1, 3–26.

ORIGIN OF LANGUAGE. 2004. Origin of language — Wikipedia, the free encyclopedia. [Online; accessed 02-11-2016].

OVCHINNIKOVA, E. 2012. *Integration of world knowledge for natural language understanding*. Vol. 3. Springer Science & Business Media.

PANG, B., LEE, L., AND VAITHYANATHAN, S. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*. Association for Computational Linguistics, 79–86.

PONZETTO, S. P. AND STRUBE, M. 2006. Exploiting semantic role labeling, wordnet and wikipedia for coreference resolution. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*. Association for Computational Linguistics, 192–199.

PRADHAN, S., ELHADAD, N., CHAPMAN, W., MANANDHAR, S., AND SAVOVA, G. 2014. Semeval-2014 task 7: Analysis of clinical text. *SemEval 2014 199,* 99, 54.

PRADHAN, S., ELHADAD, N., SOUTH, B. R., MARTINEZ, D., CHRISTENSEN, L., VOGEL, A., SUOMINEN, H., CHAPMAN, W. W., AND SAVOVA, G. 2015. Evaluating the state of the art in disorder recognition and normalization of the clinical narrative. *Journal of the American Medical Informatics Association 22,* 1, 143–154.

PUSTEJOVSKY, J. 1991. The generative lexicon. *Computational linguistics 17,* 4, 409–441.

RAHMAN, A. AND NG, V. 2011. Coreference resolution with world knowledge. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 814–824.

RESNIK, P. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 1*. IJCAI'95. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 448–453.

ROACH, E., LLOYD, B. B., WILES, J., AND ROSCH, E. 1978. Principles of categorization.

RUPPENHOFER, J., ELLSWORTH, M., PETRUCK, M. R., JOHNSON, C. R., AND SCHEFFCZYK, J. 2006. Framenet ii: Extended theory and practice.

SABOU, M., DAQUIN, M., AND MOTTA, E. 2008. Exploring the semantic web as background knowledge for ontology matching. In *Journal on data semantics XI*. Springer, 156–190.

SÁNCHEZ, D. AND MORENO, A. 2006. Discovering non-taxonomic relations from the web. In *International Conference on Intelligent Data Engineering and Automated Learning*. Springer, 629–636.

SAVOVA, G. K., MASANZ, J. J., OGREN, P. V., ZHENG, J., SOHN, S., KIPPER-SCHULER, K. C., AND CHUTE, C. G. 2010. Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications. *Journal of the American Medical Informatics Association 17,* 5, 507–513.

SCHUTZ, A. AND BUITELAAR, P. 2005. Relext: A tool for relation extraction from text in ontology extension. In *International semantic web conference*. Vol. 2005. Springer, 593–606.

SHARMA, A., SWAMINATHAN, R., AND YANG, H. 2010. A verb-centric approach for relationship extraction in biomedical text. In *Semantic Computing (ICSC), 2010 IEEE Fourth International Conference on*. IEEE, 377–385.

WINOGRAD, T. 1972. Understanding natural language. *Cognitive psychology 3,* 1, 1–191.

WU, Z. AND PALMER, M. 1994. Verbs semantics and lexical selection. In *Proceedings of the 32Nd Annual Meeting on Association for Computational Linguistics*. ACL '94. Association for Computational Linguistics, Stroudsburg, PA, USA, 133–138.

XIAO, C., ZHENG, D., YANG, Y., AND SHAO, G. 2009. Automatic domain-ontology relation extraction from semi-structured texts. In *Asian Language Processing, 2009. IALP'09. International Conference on*. IEEE, 211–216.

ZABLITH, F. 2009. Evolva: A comprehensive approach to ontology evolution. In *European Semantic Web Conference*. Springer, 944–948.

ZHANG, D. AND LEE, W. S. 2003. Question classification using support vector machines. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*. ACM, 26–32.