Author Identification in Free Texts

Kay Wang

A thesis submitted to Auckland University of Technology

in partial fulfillment of the requirements for the degree of

Master of Computer and Information Sciences (MCIS)

2019

School of Engineering, Computer and Mathematical Sciences

Attestation of Authorship

I hereby declare that this submission is my work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgments), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Signature: Date: 15 July 2019

Acknowledgment

This thesis was completed as part of the Master of Computer and Information Sciences (MCIS) course at the School of Computer and Mathematical Sciences (SCMS) in the Faculty of Design and Creative Technologies (DCT) at the Auckland University of Technology (AUT) in New Zealand.

First and foremost, I would like to sincerely thank my parents for the financial support they provided during my entire time of academic study in Auckland. Second, I would like to thank my sister Ivy Wang and all my friends for their support and companionship. Finally, I intend to express my sincere gratitude to my supervisor Dr Parma Nand. This work would not be accomplished without his instruction and encouragement. Also, I want to thank the school administrators for their support and guidance through the MCIS in the past years.

Abstract

Information Extraction is a popular topic in the Natural Language Processing area. This thesis focuses on author identification in free text. This study divided the author identification task into two subtask, quotation extraction and speaker attribution. The entire system contains two parts, a rule based model for quotation extraction and a machine learning model for speaker attribution. The resource domain used in this thesis is the literary narrative. There is also a generalisation test on the news domain. The results of the experiment show that the rule based model can achieve a 0.88 F-score on quotation extraction and the best result of a machine learning model is 85.7% accuracy. The overall test on the entire system returns 77.9% accuracy on the literary source domain and 73.6% on the news domain.

Keywords: Natural Language Processing (NLP), Author Identification, Quotation Extraction, Speaker Attribution, Conditional Random Field (CRF), Support Vector Machine (SVM)

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Language is indispensable in human's lives, and usually is used to exchange information between humans. It is a carrier of information and can be used like coding. When the author writes the text, the information is expressed or encoded in textual form. The result of the encoding is presented as a string of words. The readers can then use the same language as a decoding method to get the information from the authors. Natural language processing (NLP) is utilised in programming computers to decode the information in human languages. In order to achieve this kind of communication between machines and people, a large amount of natural language data may be processed and analysed to extract information. Information extraction is one of the main

tasks of Natural Language Processing. It also contains several subtasks, such as Named Entity Recognition (NER), Coreference Resolution, Event Extraction, and Relationship Extraction. This thesis will study Quotation Extraction and Speaker Attribution in free text.

1.1 Background and Motivation

The personality of speakers can be recognised by other people according to what they say. The study of personality and speech can be traced back to the 1930s (Chapple, 1940; R. Ramsay, 1966; Taylor, 1934). These researches analyse people's speeches and classify the personality of the speakers. Ronald W Ramsay (1968) studied the connections between personality and the non-content aspects of speech, such as the length of sounds, the silence in the speech, units and the sound or silence ratio. This research shows from what the speakers said and how they said it, that it is possible to determine the personality of the speakers. Currently, with today's technology, it is not quite possible to get the machine to understand human languages as we do, but computers can provide different perspectives by analysing large amounts of data. In most natural language text, it is not possible to know these kinds of

non-content aspects of speech, but the content of quotes and utterances can still show much of the personality of the speaker. Earlier researches in the area of personality recognition, considered conversation to be an important feature. For example, Mairesse and Walker (2006) used conversations to identify the speakers' personality in texts. In 2007, the same group used linguistic cues to recognise personality from conversation and text (Mairesse, Walker, Mehl, & Moore, 2007).

Quotation and utterance analysis can not only be used for personality recognition, but the conversation between people also can show the relationship between them, according to Salamin, Vinciarelli, Truong, and Mohammadi (2010). It is reasonable to assume that two people have a relationship if they talk to each other. In the field of social action and human interaction, the analysis of conversation has long been studied (Atkinson, Heritage, & Oatley, 1984). Compared to human perspectives, the analytical results are more accurate and unbiased (Muhuri, Chakraborty, & Chakraborty, 2018).

Salamin, Favre, and Vinciarelli (2009) stated that attributing the participants of conversation would be useful for social network extraction. There are various research focused on interaction and social networking among characters in literary narratives. For example, Elson, Dames, and McKeown (2010)

used conversation among characters to extract the social networks from literary fiction. The first extract of this research on the conversation networks in literary fictions, then uses the result of the conversation network to construct a social network for the characters. This research also mentioned that the effective methods used for author identification can provide some different understanding of social networks in literary narratives. These new understandings could also be used for more in-depth analysis.

More recently Muhuri et al. (2018) extracted the social network of the characters from Bengali literature. The social networks in fictional literature have also been studied (Dekker, Kuhn, & van Erp, 2018). In this research, the utterances of characters also played an important role. In the research of He (2011), the patterns of character connections have also been researched while doing the analysis on the social network in novels. These patterns could be used for deeper analysis of social connections in literary narratives.

Author identification can also be useful in the creation of audio books. Early research (Zhang, Black, & Sproat, 2003) worked on quote attribution, in order to build a speech synthesis system which could use different voices to read the parts of quotations in children's stories.

These previous studies have shown that this topic is worth studying and

there is value in it continuing to be studied. This thesis studies quotation extraction and author identification in free text, and the generalisation of the performance of different methods are also a focus.

1.2 Research Question

The main purpose of this thesis is to investigate machine learning approaches which can be used to extract the author information and what they said in free text. To identify which author is doing the talking, first it should be determined whether there is speech. Then, after a conversation is detected in the sentence or sentences, the spoken content and the author should be extracted. If the author is represented by a pronoun, the coreference resolution should be done to find the author's name. Hence, the research questions of this thesis are:

Question 1. *How are quotes extracted in the text?*

The utterances of a person in text can exist in different formats such as direct or non-direct speech. In this thesis three types of quotation will be considered. The main method used to extract quotes is regular expression.

Question 2. *How to identify the correct author with each quote?*

Recently, the most popular ways to detect author attribution were: grammar-based, the machine learning approach and deep learning approach. In this thesis, machine learning approaches are chosen to do author identification.

Question 3. *Which machine learning approach can be used to identify the author?*

Author identification can be seen as a classification problem. It can be a binary classification; for each author-quote pair returned is either true or false. It can also be a multi-classification problem; for each quote find the most likely author match from all the authors in the text. Thus, two different machine learning approaches will be discussed in the next chapters.

1.3 Contribution

This section clarifies the contributions of this thesis. In this thesis, author identification using machine learning is proposed and designed. The key contributions of this thesis are shown below:

- Investigate and summarise the relevant work of quotation extraction and author identification.
- Develop a rule based method for quotation extraction.

- Compare the results of two different machine learning models, Conditional Random Fields (CRF) and Support Vector Machine (SVM) Rank.

- Evaluate the author identification system on two corpuses from different source domains.

- Analyse the results of the experiment for future improvement.

1.4 Objective of this Thesis

First of all, this thesis introduces author identification using machine learning which could be attributed to the author in the text. Machine learning is a new field in artificial intelligence. The principles of machine learning are demonstrated and evaluated in this thesis.

Secondly, the overall objective of this thesis is to develop a system of author identification suitable for free text. In order to identify the author of each quotation in free text, the advance request is to extract the quotations from the text. Thus, this goal is mainly divided into two subtasks, quotation extraction and speaker attribution. In this thesis, a rule based approach is built to do the quotation extraction task. For the speaker attribution task,

the two machine learning models that will be used are: CRF and SVM.

Finally, in order to achieve the author identification in free text, a cross domain test on the system is done to evaluate the generalisation performance. Therefore, to compare experiment results from those models, it is necessary to find the best one that is suitable for quotation extraction and speaker attribution.

1.5 Research Structure

This thesis is divided into six chapters. The content of each chapter is as follows:

Chapter 2 includes the literature review of previous related works. The background knowledge of quote extraction and speaker attribution are both the focus of this chapter. For each task, both rule based and machine learning approaches are contained. Some works based on deep neural networks are also mentioned.

Chapter 3 presents the methodologies used in this thesis. In this chapter, the methods used for quote extraction are discussed, and the fundamental knowledge of CRF and SVM Rank systems are introduced. This chapter also

presents the experiment design and implementation of the research.

Chapter 4 shows the results and outcome of the experiments. This chapter discusses the details about the results of our experiments assisted by using tables. At the end of this chapter, the limitations of the research experiment are addressed as well.

Chapter 5 discusses the findings. According to the results in Chapter 4, this chapter analysed the errors made by the system. The comparison of two machine learning models used for speaker attribution will also be discussed in this chapter.

Finally, Chapter 6 summarises the thesis. This is the final chapter of this thesis, which includes the summary of the work and the future work of this research.

Chapter 2

Literature Review

This chapter reviewed some literature related to the topic of the thesis. The literature review first started with a knowledge of linguistic studies about quotations and dialogues. It continues by discussing some related works about quotation extraction and speaker attribution. These related works cover a variety of different domains and methods.

After the review of related works, the algorithms used in this thesis are introduced. This thesis used three machine learning algorithms. A Hidden Markov Model (HMM) is used to detect name entities in the texts. Two machine learning models are used for speaker attribution, CRF and SVM.

2.1 Linguistic Study

In order to do the quotation extraction and speaker attribution task, an analysis of the language structure is necessary. Rather than syntactic analysis in other research, Sagot, Danlos, and Stern (2010) analysed the structure of quotations at the discursive level in French. Their research focused on direct quotations in speech headed by a speech verb. This section includes the structure of quotations and the categories of dialogue. In the dialogue categories part, the patterns used in previous research are discussed.

2.1.1 Quotation Structure

Elson and McKeown (2010) stated that quotes can be considered as a block of text between quotation marks. However, in addition to text between quotation marks, they can also exist in different forms in the text. Scheible, Klinger, and Padó (2016) listed a series of difficulties in the quotation extraction task. This includes some which have quotations represented only by lexical prompts without quotation marks; the length of each quotation is arbitrary. In several quotation extraction researches (Krestel, Bergler, Witte, et al., 2008; O'Keefe, Pareti, Curran, Koprinska, & Honnibal, 2012), quotations that usually appear in texts are divided into three types, direct quotes,

indirect quotes and mixed quotes.

Direct Quotes

Direct quotes commonly appear between either single or double quotation marks. This kind of quote shows exactly what the author said. Here is an example from Chapter 1 of the novel Jane Austen's *Emma*:

> *"Poor Miss Taylor!–I wish she were here again. What a pity it is that Mr. Weston ever thought of her!"*

However, in news articles direct quotations only represent a limited portion. Pareti, O'Keefe, Konstas, Curran, and Koprinska (2013) stated that in the Penn Attribution Relation Corpus (PARC), that only around 30% of quotations appear in direct quotation type. And the portion of direct quotation in the Sydney Morning Herald Corpus (SMHC) is 52%.

Indirect Quotes

Indirect quotes do not appear between or contain quotation marks; it is usually an alternative to direct quotation. The words in the indirect quotes are not exactly what the person said. This kind of quote is usually a paraphrase

or a summary of the original words from the speaker. Here is an example from the Reuters Corpus:

The offices will be staffed by employees who will bring specific expertise in their country's banking system to each marketplace, the companies said.

This type of quotation usually appears in the field of news, when a reporter outlines a speech from another person but does not report the content of the speech verbatim. There are two advantages of using indirect quotation in the news domain. Firstly, if the original speech is too long, the journalist could use indirect quotes to reword it for brevity. Another advantage is that journalists can remove the disfluencies in the original speech by rewording the speech as needed. The extent of indirect quotes is not marked by quotation marks as direct quotes, so the reader needs to determine the extent of the speech themselves.

Mixed Quotes

Mixed quotes contain a directly quoted portion and an indirect portion. Here is an example from the Reuters Corpus.

A Stock Exchange of Singapore spokesman said the company had requested

for the suspension pending an announcement "soon".

The last type of quote that needed to be extracted in this thesis are mixed quotes. Mixed quotes are formed by two parts, the direct part and the indirect part. This means that part of these quotes appears within the quotation marks, and another part of the quotes appears outside the quotation marks. It is typically understood, that the direct part is recorded verbatim, while the part outside the quotation marks is reworded.

2.1.2 Dialogue Category

Sarmento and Nunes (2009) built a system with 19 different patterns to extract quotes from online news feeds. The structure of their patterns are similar as "[Optional Ergonym][Speaker Name][Speech Verb][Direct or Indirect Quote]." For these patterns, the speech verb is from 35 selected verbs and the name of the speaker should be fully mentioned. They also mentioned that these 19 patterns and 35 speech verbs are not enough for extracting all different kinds of quotations. In order to build a robust and generalised rule based approach for quotation extraction, the context of quotes is classified into some more generalised categories. The following dialogue categories are extended from Elson and McKeown (2010).

- **Name Trigram :** This category of quotation appears consecutively with the speaker name and a speech verb. Due to the different permutations of speech verb and speaker name, in this category there are six subcategories. Here is one example of this category from the Reuters Corpus:

  "In our first release we won't support the Microsoft browser, but we expect them to be a part of our service shortly," Dodd said.

  In this example, "said" is the speech verb and "Dodd" is the speaker name. Thus, the pattern of this sentence is [Quote][Speaker Name][Speech Verb].

- **Anaphora Trigram :** This quote category is similar to Name Trigram, except that the speaker of the quotation is shown as a pronoun. There are also six subcategories here in different permutations, such as Name Trigram. Although this category of quotation can not show the speaker's name directly, the gender of the speaker (male, female or plural) can be decoded through the pronoun. Here is an example from the novel Jane Austen's *Pride and Prejudice*:

"I am sick of Mr. Bingley," cried his wife.

The pattern of this sentence is [Quote][Speech Verb][Pronoun]. From the pronoun "his wife," it can be speculated that the author of the quote is a female.

- **Added Quote :** This category of quotation appears in the same paragraph after another quote. The preceding word of Added Quote is usually a speech verb, pronoun or the name of the speaker. Here is one example for this category from the Reuters Corpus:

  "In the upcoming months you will see all sorts of integrated activity," Dodd said. "We've been in the Internet, in fact profitable and making money in the Internet business, longer than anybody."

  In this example, the preceding word of the Added Quote is the speech verb "said" and it has the same speaker as the proceeding quotation.

- **Conversation :** This category of quotation usually appears in a paragraph on its own. The preceding or following paragraph of this target quote should also be a quotation. Usually in a conversation, there are only two participants. The speakers of two neighbouring quotations are

not the same and the previous and next quotation have a high chance of having the same speaker. In some situations, it is possible that there are more than two participants in one conversation. This may provide a noise source for the machine learning models.

- **Quote Alone :** This category of quotation appears by itself in a paragraph. It is similar to the conversation but the preceding and following paragraphs are not quotations.

2.2 Related Work

This section presents an amount of preliminary work that is highly relevant to the quotation extraction and speaker attribution task. For the quotation extraction task, the kinds of quotations the previous research focused on, and the resource domain used in author identification research are both discussed. Some research studied these two tasks at the same time and they provided only the overall performance of their system.

2.2.1 Quotation Extraction

The purpose of the quotation extraction task is to find or locate all the spans that represent quoted speech in the text (Scheible et al., 2016). Currently, most approaches that are used to achieve this purpose are rule based. There are some approaches however, using machine learning models.

Recently, Smeros, Castillo, and Aberer (2019) used regular expressions based on word patterns which are manually defined through an information extraction model to extract quotes from science news. This research also tried to use syntactic patterns combined with quotation marks to do the extraction, but the results of simple regular expressions without the word patterns are not good enough.

Krestel et al. (2008) uses a grammar based system with six general lexical patterns to detect and attribute the quotation in reported speech. This research mainly focuses on the syntactic markers and speech verbs. All types of quotations are considered. Their test set is a small subset of the Wall Street Journal which only contains seven articles; the total number of quotations in the test set is 133. They evaluated this test set and achieved 99% and 74% recall.

The EVRI portal provides a Quotation Extraction API for English news

feeds (Liang, Dhillon, & Koperski, 2010). The approach of this system is based on rules and it uses several linguistic techniques such as entity recognition, coreference resolution and disambiguation which is automatically provided by standard auxiliary processors. These Quotation Extraction APIs already have more than 10 million quotes extracted from English news feeds, and about 60 thousand new quotations that extracted from about 50 thousand news feeds are added in one day. This system can only extract text within the quotation marks, which are direct quotations and the quoted part in mixed quotations. Indirect and the other parts of mixed quotations are ignored. In this paper, the performances of this system such as precision, recall and accuracy are not mentioned.

O'Keefe et al. (2012) used some simple regular expressions to extract quotes and achieved over 99% accuracy on three corpora. The focus of this research is quotation extraction on large scale news data. They used three large corpuses with a total of 11,584 quotations. This research only extracted the quotes between quotation marks and the multi-paragraph quotes between quotation marks, which are direct quotations and the direct portion of mixed quotations.

Pareti et al. (2013) developed the state of the art sequence labeling ap-

proach which focuses on the extraction of indirect quotes. This study used political and literary linguistic corpora to train their supervised machine learning models. They presented two supervised approaches; one is the token based approach and the other one is the constituent based approach. The token based approach has better performance which can achieve 92% precision and 86% recall.

Quotations can be found in various configurations. Besides the English corpus, there has also been research work on extract quotation in other languages especially in the news domain. Pouliquen, Steinberger, and Best (2007) present a fully functional software which can identify direct speech quotations in eleven different languages. This software system is based on lexical rules. In some applicable situations, their software can detect the speaker of the extracted quotation at the same time. They have also done the coreference resolution; even the speakers name is spelled in different ways, and the quotes can still be assigned to the same person. Although this system can deal with eleven different languages with high precision, however, on multilingual text the recall of this system is low.

de La Clergerie et al. (2009) presented a system called Sapiens based on syntactic rules. They used a large French news wire corpus, L'Agence

France-Presse. For detecting direct quotes, they only use simple rules, and for indirect and mixed quotations the system used a deep linguistic processing chain. They manually collected 144 quotation verbs as the cues of the detection. The text was first parsed, and then verified if the verb was one of the quotation verbs. To evaluate the system, they manually labeled 40 quotes from 40 different news articles and the Sapiens system correctly detected 32 of them. Although the results achieved 80% recall, the test set is manually selected and might be biased towards structures recognised by the system.

Syaifudin and Nurwidyantoro (2016) use the rule based method to identify quotations from Indonesian online news texts. In this research, the feature used in formulating class rules is the presence of entities and reporting verbs in the sentence. They built the rules of their system based on the pattern structure of quotations in news. The precision of their system is 99.013%, the recall is 79.936%, and the overall accuracy is 88.618%.

Salway, Meurer, Hofland, and Reigem (2017) used a statistical dependency parser, a few regular expressions and a look-up table to extract quotes from Norwegian newspapers. Their approach achieved 97.8% precision, but the recall is 57%.

Authors	**Approach**	**Language**	**Type**	***Precision***	***Recall***
Pouliquen, Steinberger, and Best (2007)	Rule Based	Eleven Languages	D	99.2%	-
Krestel, Bergler, Witte, et al. (2008)	Grammar Based	English	D,I,M	99%	74%
de La Clergerie et al. (2009)	Rule Based	French	D,I,M	99%	80%
Liang, Dhillon, and Koperski (2010)	Rule Based	English	D	-	-
Pareti, O'Keefe, Konstas, Curran, and Koprinska (2013)	Supervised Learning	English	D,I,M	92%	86%
Syaifudin and Nurwidyantoro (2016)	Rule Based	Indonesian	D,I,M	99%	79.9%
Salway, Meurer, Hofland, and Reigem (2017)	Grammar Based	Norwegian	D,I,M	97.8%	57%
Smeros, Castillo, and Aberer (2019)	Rule with Pattern	English	D,I,M	-	-

Table 2.1: Related work on Direct (D), Indirect (I) and Mixed (M) quotation extraction in different languages.

To summarise the literature reviewed in this section, Table 2.1 shows the related work on quotation extraction. The results of these researches are not directly comparable, because they evaluated their approach using different test sets and the language studied by these researches are not same. The results of these previous researches show that rule based approaches can achieve high precision but low recall in the quotation extraction task. However, it works better on the extraction of direct quotes than other kinds of quotations. This phenomenon is due to the fact that quotations can have various types of syntactic forms (Elson & McKeown, 2010). Rule based approaches can perform well on news text, but can also perform well on other types of text. Unfortunately, in free text, the quote may exist in any form. Machine learning has an approach without defining the template or patterns manually.

2.2.2 Speaker Attribution

The previous section discussed the literature on the quotation extraction task. This section will focus on the research on speaker attribution. Some researches studied both speaker attribution and quotation extraction. For those researches, the speaker attribution part of their work will be discussed

in this section, and some other work on the speaker attribution task also will be discussed.

In the field of speaker attribution, the early works are focused on the domain of children's stories. Zhang et al. (2003) state a rule based approach for speaker identification, in order to build a text to speech system and read quotations from different speakers in different voices. They analysed their corpus to develop rules for speaker identification and found that the accuracy of their system is between 47.6% and 86.7% in different test documents. The rules used to identify speakers are based on the corpus, which makes these rules not widely applicable.

Another work expands on the approach provided by Zhang et al. (2003). The work provided by Mamede and Chaleira (2004) uses a hand-crafted decision tree to attribute speakers in Portuguese children's literature. The approach they provided uses five rules based on the structure of quotations in the literature, achieving 65.7% accuracy. However, the test set they used to evaluate their approach is a small corpus which has only 35 quotations.

Glass and Bangay (2007) use naive scoring techniques with some simple rules to attribute the speaker of quotes in children's stories. The focus of this study is to find the connection between quotations, speech verbs and

verb agents. Once they are able to recognise the verb, they will extract the speech-verb-actor link from the sentence, and then they use this link mode to exploit the potential information of the recognition task. Their work is based purely on manual coding rules to implement a scoring scheme. Their method produces 79.4% accuracy in a corpus of manual annotated children's stories. Another study proposed by the same authors use a rule generalisation method (Glass & Bangay, 2006). In this work, they use the seed rule set to generate new extra rules by adopting a merge scheme. The seed rule set is first constructed in a tree form, and then based on the merge scheme, a new hierarchical rule set can be generated.

Iosif and Mishra (2014) adopt a more sophisticated approach. Their system first detected speech verbs by using some scoring techniques with simple features. Then for each speech verb, the actor is identified based on their defined rules. In this research, they also considered that the actor could be a common noun or not present. This system is evaluated on 17 different stories with 554 quotes, and the average accuracy of their evaluation is 84.5%. In addition to speaker attribution, this research also attributes gender, age and personality for each character. Although all of these four researches worked on a children's story domain, they used different data to evaluate

their system. Therefore, the results of these researches cannot be compared to each other.

Elson and McKeown (2010) use machine learning methods for quote attribution. They provide a supervised statistical learning solution for the speaker attribution task. The corpus they used in their research is constructed by 11 classic literature narrative works. They first construct a feature vector for each quote-speaker pair, and then use the machine learning models to construct the speaker identification system. The statistical models will give each speaker candidate a binary label and a probability score. In the final phase, their approach uses a reconciliation scheme to group all the results together and select the speaker with the highest probability score provided by the model. The overall accuracy of this system is 83%. This system has been used in another work by the same author to extract social networks in novels (Elson et al., 2010).

Another domain of concern for the studies on speaker attribution task is news. Pouliquen et al. (2007) provides a news monitoring system that uses multi-language pattern matching to process news articles. This system is able to attribute speakers for tens of thousands of news articles in one day. They have designed a very high precision system because the same quotations may

appear multiple times in different news providers and different languages. However, due to the inherent redundancy of the data, the rate of low recall is greatly increased. In order to focus on high precision, their system ignores the use of pronouns and only attempts to attribute quotes made by known 50,000 speakers. The elements such as quotation marks, people's names, and speech verbs, can be found in the text through their system. After that, they check the manually defined list to make sure that if those elements are matched with any set of patterns. Therefore, the precision of their system reaches 99.2%.

The work of Sarmento and Nunes (2009) is similar to the system provided by Pouliquen et al. (2007). However, their research only focuses on one language. Regardless of the structure of other languages, they can use more precise patterns. The system proposed by Sarmento and Nunes (2009) is called Verbatim, which is a pattern matching system that can extract speech quotations from Portuguese news articles. Through studying the typical patterns of speech, they defined 19 different variations which are similar to the typical speaker pattern. After that, they put 35 verbs in a candidate pool as the candidate speech verb. Therefore, based on syntactic rules, this matching system can finish the identification work of the speaker following

those 19 different variation patterns and the speech verb candidate pool. This system achieved a precision of 98.2% in the speaker attribution part. Although the precision is high, the system only attributes the speakers of the quotes they extract. From 26,266 news articles only 570 quotes are extracted, which suggests that the recall of this system is low.

Schneider et al. (2010) developed a system PICTOR. This system is designed to extract, attribute and visualise quotations from English news articles. They manually designed a Context Free Grammar (CFG) for both quotations and speakers with 273 nonterminal rules. The CFG is able to recognise direct and indirect quotations, and attribute the speakers for these quotes. They did not provide accuracy results for the quotation extraction task and speaker attribution task separately. The overall performances of their approach achieves 75% precision and 86% recall by allowing partial matching. When considering a perfect match, they have a precision of 56% and a recall of 52%.

While data sparsity is always a problem in supervised learning, it may be necessary to use unsupervised learning approaches for speaker attribution. Pavllo, Piccardi, and West (2018) leverage the redundancy of popular quotes for building unsupervised bootstrapping models. This research focuses on

the attribution of the same quotation with different contexts in large news corpora. Their approach follows fully unsupervised paradigm and can achieve 90% precision and 40% recall.

Celikyilmaz, Hakkani-Tur, He, Kondrak, and Barbosa (2010) also provided an unsupervised learning method for speaker attribution. This research presented an Actor-Topic Model (ACTM) to identify the speakers of converesation in literary narratives. ACTM is a generative model that extends the author-topic model (Rosen-Zvi, Griffiths, Steyvers, & Smyth, 2004). It used an unsupervised learning method to extract dialogues and actors from literary text. The performance of this model is measured against the gold-standard by a statistical measure, mean reciprocal rank (MRR). The accuracy of this model is from 44.5% to 65.2% using a different dataset. After identifying the speaker of the conversations, they also constructed the social networks of the characters in novels.

There are other studies that focus on areas that differ from the studies previously presented. Yeung and Lee (2017) were the first to study the evaluation of listener identification. The domain of this study is literary text. They trained a CRF model with CRF++ (Kudo, 2005) to identify the speaker and listener for direct quotation. Their system achieved an accuracy

of 52.46% to 66.09% of speaker identification and 28.46% to 56.97% of listener identification based on the different datasets.

In summary, there is a series of studies on citation attribution that differ greatly in the methods they use. Usually a rule based approach can provide high precision. However, rule based approaches have problems with coverage and generalisation. Therefore, a method based purely on manual coding rules is not feasible. Machine learning methods can automatically learn implicit rules in statistical form without human effort. Another advantage of machine learning methods is that they can be better extended to new domains.

2.3 Machine Learning

The formal definition of Machine Learning is that for some tasks "T" a computer programme can learn from experience "E" to improve the performance "P" (Mitchell et al., 1997). Rule based programmes need to manually craft all the rules and then enter the data to get the output. The machine learning method only needs to input data, sometimes with labels, to train the model, and then a model for this task is generated.

Machine learning algorithms can be classified into Supervised Learning,

Unsupervised Learning and Semi-supervised Learning. In supervised learning, the dataset used to build a model should contain both the inputs and the labels which are the desired outputs. For unsupervised learning algorithms, the dataset used to train a model does not need to be labeled. Semi-supervised machine learning algorithms sit between supervised and unsupervised learning because they train a model with both labeled and unlabeled data. The amount of unlabeled data is usually larger than labled data because it is easier to obtain.

There is another way to classify machine learning algorithms based on the output they provided. In this way, machine learning algorithms usually are classified in two categories, Regression and Classification. The outputs of regression algorithms are continuous, which means they can have any value within this range. The output of classification algorithms are a limited set of values, and the result can only be one of this set of values.

Machine learning can efficiently process large amounts of data. Although it usually provides faster, more accurate results, it may require additional time and resources to train properly. The selection of machine learning algorithms are usually based on the purpose of the task and the dataset.

2.4 Hidden Markov Model (HMM)

The Hidden Markov Models (HMM) differs from the ordinary statistical Markov model in the unobservable state of its modeling system. In Hidden Markov Models, states are not directly visible, but state-dependent outputs are visible (Eddy, 1996). HMM is widely used in natural language processing especially for the problems based on time series or state sequences. Figure 2.1 shows an example of the HMM used in natural language processing (Seymore, McCallum, & Rosenfeld, 1999).

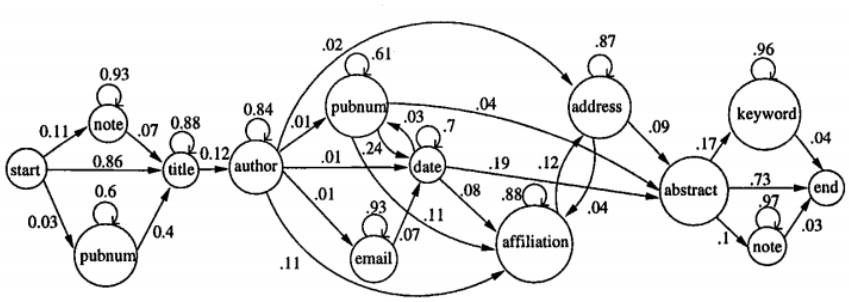

Figure 2.1: Example of the HMM used in NLP.

The HMM model has two hypotheses: the homogeneous Markov chain hypothesis, and the observation independence hypothesis (Matteucci, 2008). First assumption of HMM model is that the hidden state is only relevant to the hidden state before it. However, in practical applications, as shown in

Figure 2.1, a word in a sentence may be related to two or three words.

2.5 Conditional Random Fields (CRF)

Conditional Random Fields (CRFs) is a machine learning model and has some similarities with HMM (Ng & Jordan, 2002). The main difference between Linear-CRF and HMM is that, the HMM model finds the joint distribution $P(x, y)$ while the Linear-CRF model wants to obtain the conditional probability $P(y|x)$ (Christian Nørgaard Storm Pedersen, 2012). There are several different types of CRF model. In this thesis, only linear-CRF will be discussed because it is the main principle of the CRF model. The main usage of the Linear-CRF model is to solve three specific problems which are: training, assessment and decoding (Qi & Chen, 2010).

2.5.1 CRF Suite

There are various implementations of CRFs. In this thesis, CRFsuite (Okazaki, 2007) is chosen to be used in the experiment. The CRFsuite implemented the Linear-chain (first-order Markov) CRF. The CRFsuite can help the users to train and use CRF models in an easy and fast way. The data format in

the CRFsuite is quite simple, each line consisting of a label and attributes of an item, consecutive lines representing a sequence of items and an empty line denotes an end of item sequence.

CRFsuite implements many state of the art training methods, such as Limited-memory BFGS (Nocedal, 1980), Stochastic Gradient Descent (Shalev-Shwartz, Singer, Srebro, & Cotter, 2011), and Adaptive Regularization of Weight Vector (Mejer & Crammer, 2010), and so on. After training a CRF model with CRFsuite, the performance evaluation is also quite easy. It can output the measurements such as precision, recall, F1 scores of the model. Besides these features, CRFsuite supports many different programming languages. It provides support interface for various languages such as Python, C++, and so on.

2.6 Support Vector Machine (SVM)

Support Vector Machine (SVM) is a classification and regression machine learning method introduced by Vapnik (1999). The SVM models have a special property that can simultaneously minimise empirical errors and minimise geometric edge regions; therefore they are also known as maximum interval

classifiers. SVM is based on the Structural Risk Minimization (SRM) principle and the statistical learning theory, and belongs to a family of generalised linear classifiers.

The basic idea of SVM is to be a solution for a single hyperplane that divides the training data set correctly and has the largest geometric spacing. For linearly separable data sets, there are an infinite number of hyperplanes that can be used to classify the training examples. However, the hyperplanes with the largest geometric spacing are unique. The best separation hyperplane can provide the maximum margin between the support vectors. Support vectors is a set of training examples that are closest to the separated hyperplane. New examples can be easily classified by checking which side of the hyperplane they fall on after finding the hyperplane. Figure 2.2 shows an example of optimal separating hyperplanes in two-dimensions.

An important innovation in support vector machines is the kernel. Kernel technology is very powerful for two reasons: first, it allows us to learn nonlinear models. Second, the implementation of kernel function k is usually more efficient than directly constructing the whole function and recalculating dot products.

The SVM algorithm was originally designed for binary classification prob-

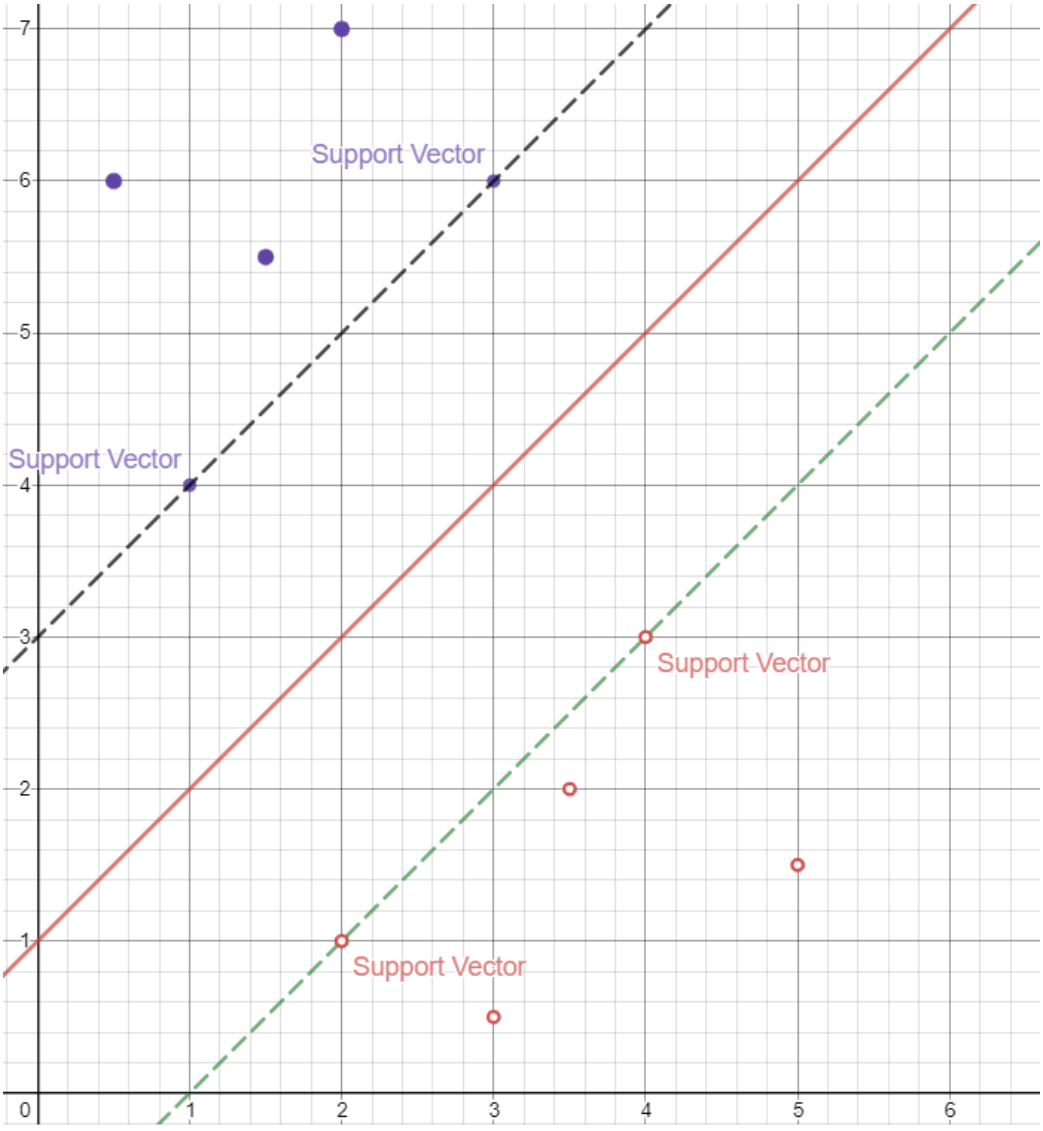

Figure 2.2: Optimal Separating Hyperplane

lems. The main way to achieve multiple classifications is to translate multiple types of problems into multiple, two-class problems. One common used method is the One-Against-All (OAA) method (Hsu & Lin, 2002). The OAA methods classifies the samples of one category into one class and all the other samples into another class so that the samples of the k categories are k numbers constructing two-class SVMs. However, the calculation of OAA methods will be large when the dataset is large.

2.6.1 Ranking SVM

Ranking has always been one of the core issues of information retrieval (Liu et al., 2009). Learning to Ranking (LTR) uses machine learning to solve ranking problems. LTR has three main methods: PointWise, PairWise, and ListWise (H. Li, 2011). Ranking SVM algorithm is a type of PointWise method proposed by Herbrich (2000). Joachims (2002) introduced a method for optimising the search engine based on users' clickthrough data using Ranking SVM.

An SVM ranking model is a discriminative model used to rank candidates based on confidence scores. To rank by using machine learning, the main idea of Ranking SVM is to turn the ranking problem into a classification

problem. The definition of a training sample is that for input samples x_1 and x_2, if the ranking score of x_1 is larger than the ranking score of x_2, then $x_1 - x_2$ is a positive sample, and $x_2 - x_1$ is a negative sample. After redefined these inputs, an SVM classifier can be trained to classify these new training samples. By converting the sorting problem into a classification problem, the ranking problem can be solved by using the common machine learning method.

Chapter 3

Methodology

In this chapter, the methodologies used in the experiment are described in detail. The CRF model is used as a binary classifier, for each quote-speaker pair it returns a True or False according to the probability. In the following sections, the SVM ranking model is used as a multi-class classifier for each quotation. It lists the rank of every possible speakers and the one with the highest rank would be identified as the author of the quotation. The results of the experiment will be discussed in the next chapter.

3.1 Research Design

This research is designed to identify the author of each quotation in free text. The experiment is divided into three modules, data preprocessing, quotation extraction and speaker attribution. Figure 3.1 is the flowchart of the entire experiment which shows the three modules that compose this experiment.

The first module is data preprocessing which was designed to prepare the text for quotation extraction. The following module is Quotation Extraction, in this module a pattern match approach is used to extract quotations from the preprocessed text. The last module is Speaker Attribution. This is the main module of the experiment, in this module two machine learning models are trained with the same feature to identify the authors of the quotations. The quotation extraction module and speaker attribution module are built separately. This means the training and testing data of the speaker attribution models are from the corpus directly, not from the results of the quotation extraction approach. However, after the machine learning models for speaker attribution are trained, the quotations that need to be attributed in the overall test are derived from the results of the quotation extraction method.

At the end of the experiment, there is an evaluation of the system. The

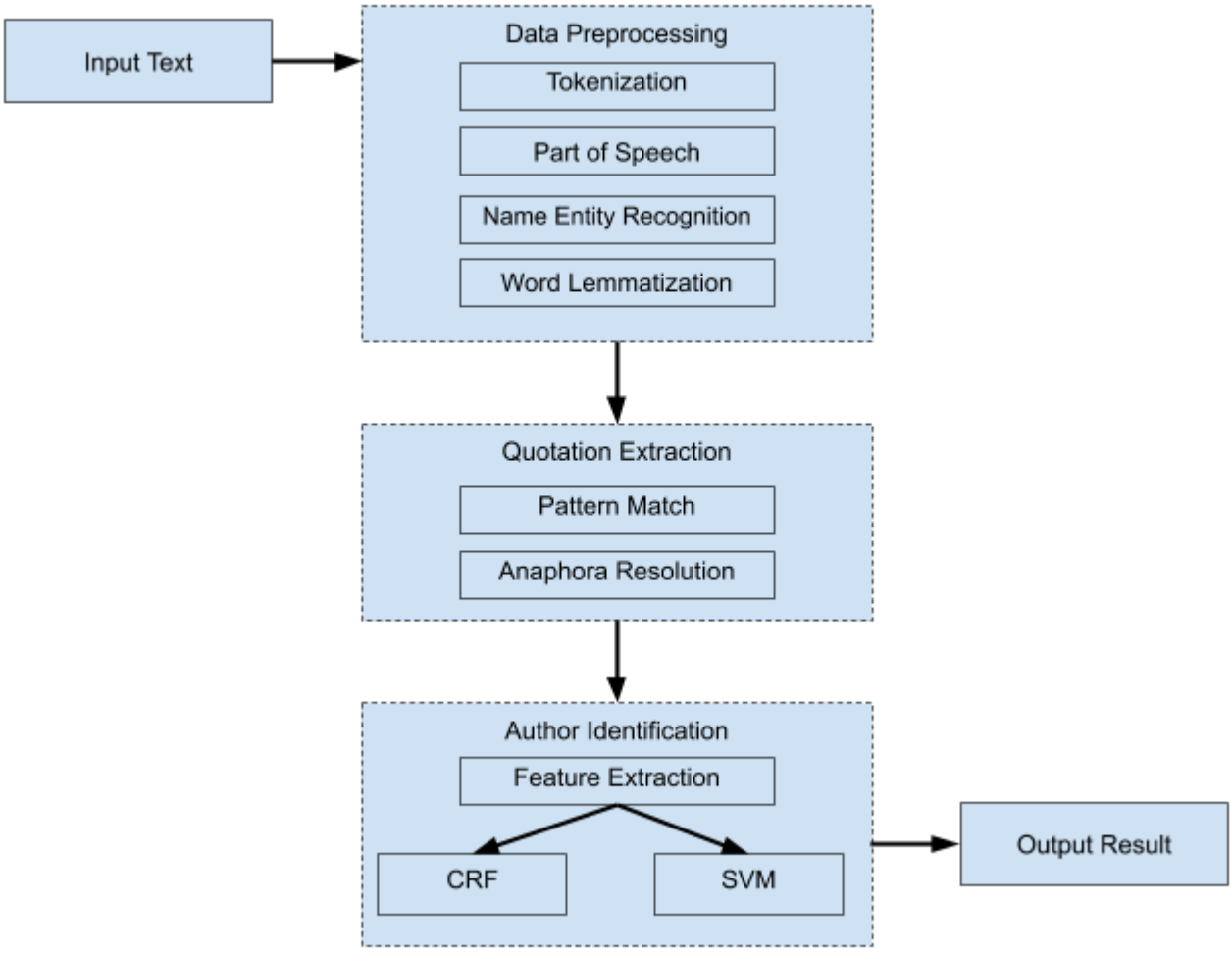

Figure 3.1: Flowchart of the author identification system.

evaluation section includes four different tests. The first is the evaluation of the quotation extraction method. The performance of the two machine learning models for speaker attribution was then evaluated, and an overall performance of the entire system was tested. Finally, the generalisation capabilities of the two models were tested separately.

3.2 Quotation Extraction

As mentioned in Chapter 2, a rule based approach with patterns could handle most situations. Thus, in this thesis a rule based approach is used for quote extraction. The rules used to extract the quotations is based on the patterns that have been explained in the previous chapter. The process of the quotation extraction step is shown in Figure 3.2.

In order to determine if there is a quotation in the sentence, a pattern matching approach is built to detect them. In the pattern matching step, 15 patterns are built based on the quotation categories discussed in Chapter 2. The main components of the patterns are named as follows: entity, pronoun, speech verb and quotation.

As the contents of a direct quotation must be enclosed in quotation marks,

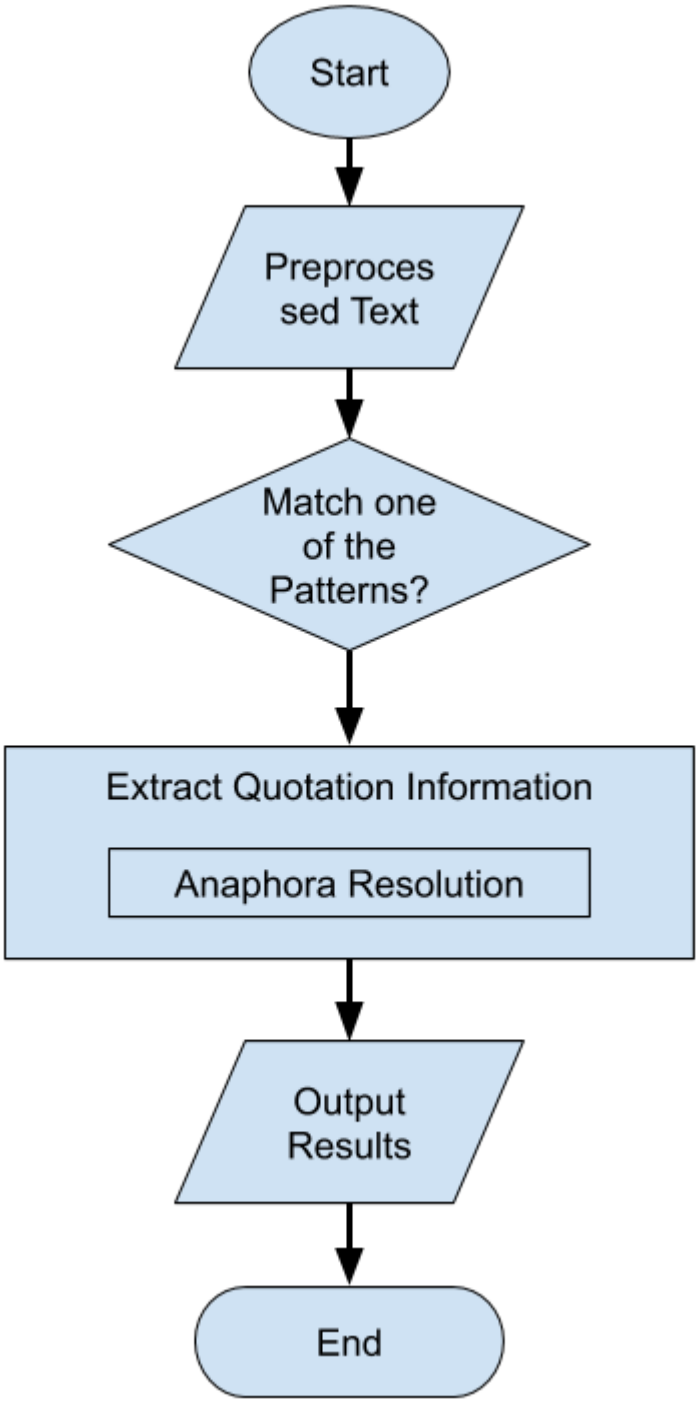

Figure 3.2: Quotation Extraction Process

the main trigger for extracting a direct quotation is the quotation mark itself. All the sentences with quotation marks are determined as quotations. Previous studies show that in a large number of literary narrative works, there are only 3.5% quoted sentences which are not dialogue text (Elson & McKeown, 2010). Since most rule based methods are affected by the low recall rate (as discussed in Chapter 2), the loss of accuracy is a worthwhile result to ensure that the recall of extracting direct quotations is warranted. In the experiment, some regular expressions are used to detect the quotation marks in sentences.

While indirect and mixed quotations are the target that need to be detected, the main trigger of this approach is the existence of the named entity, pronoun and speech verb. The name entities and pronouns have already been labeled out in the data preprocessing step.

The way to determine if it is a speech verb is by comparing the target word with the words in the speech verb lexicon. In the speech verb lexicon, all the verbs are stored in the prototype because there is a word lemmatization step when preprocessing data. The verbs in the lexicon are *advice, say, claim, tell, etc.*

For some of the quotations, the author can be identified during the ex-

traction. The results of this identification have a higher confidence than the results provided by the machine learning models in the speaker attribution part. Thus, in the overall test for the entire system, if the results of two parts are different, the results provided by the rule based approach will be selected as the final output.

3.3 Speaker Attribution

Two classifiers are used in the experiment: a CRF model and an SVM Rank model. The features used for both models are the same, which will be explained in this section below. The dataset is divided into three parts: a training set, a validation set and a testing set. The ratio of the amount of data in the training set validation set and the test set is 80:10:10. The models are first trained on the training set and then the parameters of the model are adjusted based on their performance in the validation set. In order to avoid overfitting, test sets are not used during the training and adjustment phases.

3.3.1 Feature Extraction

Before extracting the features, the context data are encoded. The basic purpose of encoding is to convert raw data and to make the features easier to be extracted. The encoding includes two main steps. First, adjectives and adverbs are removed as they are parts of speech that do not provide useful information for attributing speakers. The paragraphs or sentences which do not contain quotes, pronouns or named entities are also removed. The second step is to replace all quotes speech verbs, and speakers with symbols.

The feature set used in the experiments is an extended version based on Elson and McKeown (2010). After the data encoding, all of these features are calculated from the encoded data. For example, all the numbers in the distance feature should be the number in the encoded text, not the number in the original text. The following features are for a particular quote (Q) and speaker (S) pair.

Distance : This feature includes the distance between Q and the most recent S that appears. These distances includes the number of words, the number of paragraphs, the number of quotes, and the number of entities mentioned between Q and S.

Normalised Distance : This feature is the normalised format of the

distance feature. Due to the uncertainty of the length of the target text, this feature includes the number of words, paragraph, quotes and entity mentions between Q and the nearest S divided by the total number of words, paragraphs, quotes and entity mentions in the target text.

Paragraph : These features are derived from the 10 paragraphs preceding the quote. This includes the paragraph the quote is in, including the number of mentions of S, the number of mentions of other speakers, the number of words in each paragraph, and the number of quotes in each paragraph

Nearby : This feature includes the information of two tokens either side of Q and S. As all the adjectives and adverbs were removed in the data preprocessing step, this feature only indicates for each token whether it is punctuation, S, Q, a different speaker, a different quote, or a reported speech verb using a binary string.

Mention : This feature includes whether S or other speakers are mentioned within Q.

Quote : This feature includes the information of the target quotation Q itself. It includes the number of words in the target quotation.

3.3.2 CRF Model

The CRF model is trained in binary classification. For each quote-speaker pair, the output of the model is either "T" or "F", which is True or False. The experiment uses maximum likelihood estimation with both ℓ_1- and ℓ_2- regularisation to avoid overfitting. L1 regularisation is the sum of the weights shown in formula $\lambda \sum_{n=1}^{k} |w_i|$ while L2 regularisation is the sum of the square of the weights with formula $\lambda \sum_{n=1}^{k} w_i^2$.

The training algorithm used to train the CRF model is the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) method (Malouf, 2002). L-BFGS is a popular optimisation algorithm in machine learning, and it performs especially well when the number of variables is large. And a variant of L-BFGS, Orthant-wise limited-memory quasi-Newton (OWL-QN) (Andrew & Gao, 2007) is used for fitting the ℓ_1 regularised. The final parameters of the optimisation algorithm is shown in Table 3.1.

The parameter 'c1' is the coefficient for L1 regularisation and 'c2' is the coefficient for L2 regularisation.

The parameter 'max_iterations' is the maximum number of iterations for L-BFGS optimisation. If the iteration count exceeds this value, the L-BFGS routine terminates.

Parameters	***Value***
c1	0.1
c2	0.01
max_iterations	200
num_memories	6
epsilon	$1e-5$
delta	$1e-5$

Table 3.1: The parameters of the optimisation algorithm.

The parameter 'num_memories' is the number of finite memories that L-BFGS uses to approximate the inverse Hessian matrix.

The parameter 'epsilon' is the condition of convergence.

The parameter 'delta' is the threshold for stopping. The L-BFGS iteration stops when the log likelihood improvement of the last iteration is not greater than the threshold.

Reconciling those results and make them become a single decision for each quote is the last step of this experiment. This step is due to those results can provide binary labels and probability scores for each speaker-quote pair separately. Since the classification is independent, it may be necessary to reconcile multiple decisions as multiple speakers may be predicted for one quote. For one quotation, if there is only one speaker-quote pair, it returns the True result. The speaker would be this one.

If there is more than one speaker-quote pair returning True, we reconcile these decisions by ranking the probability. The speaker-quote pair with the highest probability is taken as the final result after these probabilities are ranked,. For each quotation, the quote-speaker pair gets the one with the highest probability.

If all the speaker-quote pairs of this quotation return False, this could

be because of two possible situations. First is that the model has classified the right speaker-quotate pair wrong. Another possible situation is that the speaker of the quotation has not been mentioned in the resource. This has rarely happened in the literary narrative corpus, but in the news corpus or other kinds of free texts, this situation is possible. Thus, there is a threshold for these outcomes; if all the speaker-quote pair's probabilities fall below the threshold, in that case the system determines the speaker of the quote has not been mentioned in the input text. The value of the threshold is the lowest probability of the true speaker-quote pair in the training set. If there is more than one speaker-quotate pair which has a higher probability than the threshold, then the speaker with the highest probability will be output as the speaker.

3.3.3 SVM Rank Model

This experiment used the SVM-rank (Joachims, 2006) to implement the SVM model. The SVM-rank is an instance of structural SVM (Tsochantaridis, Joachims, Hofmann, & Altun, 2005) and is tailored specifically for ranking issues. The SVM-rank is similar to the traditional SVM classification. It shares those similarities, however, with different outputs for different training

data input and ranking functions for special designs.

For the SVM rank model, the features used and training set are the same as the CRF model. The main difference between the training of the SVM rank model and the CRF model is that the input for SVM ranking is the set of features for all the speaker-quotate pairs for one quotation. The output of the SVM-rank model is a set of scores for each speaker-quotate pair. By using these scores, the SVM-rank model can build a global sort and directly find the speaker-quote pair with the highest score. With this model the method does not need to reconcile this step to determine which one is the final result.

3.4 Evalution

There are four aspects to the evaluation of the model performance. First, the two models are evaluated separately, and then a comprehensive evaluation tests the performance of the entire system. Finally, a generalisation evaluation of the entire system is carried out, testing the performance of this system in different areas. Two sets of experiments are done to verify the performance of the two models. So, each set comes with a result comparison between the CRF model and the SVM model. Both models are tested on

the randomly picked test set from QuoteLi3 and RCV1 as the generalisation test. To evaluate the models, the measurement used was well defined by previous studies (Sokolova & Lapalme, 2009). For evaluation, the precision, recall and F-score are defined as follows:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

- **True Positive** is the number of correct predictions when the actual output should be positive.
- **True Negative** is the number of correct predictions when the actual output should be negative.
- **False Positive** is the number of incorrect predictions when the actual output should be positive.
- **False Negative** is the number of incorrect predictions when the actual output should be negative.

The results of the experiments will be discussed in the next chapter.

Chapter 4

Results

In the previous section, the experimental process and performance evaluation have been described. This section will focus on the results of the experiment and the comparison of the CRF model and SVM model. The results of the experiment are divided into three parts. The first part is the result of the quotation extraction by the rule based approach. The second part is the test result of the two models on the test set. Finally, the result of the generalisation test of the two models are presented.

4.1 Corpora

The purpose of this thesis is to extract the speakers from free text. However, it is not possible to obtain a corpus from various fields. Therefore, this experiment mainly uses corpuses from two large domains which are the literary domain and news domain. These two domains are also the most common areas that were studied by predecessors. For literary narratives, QuoteLi3 corpus is chosen for the experiment. It is also mainly the dataset for training and testing the machine learning models. For news data, a subset of Reuters Corpus (RCV1) is used. These corpuses are used for the cross domain and generalisation test. This section will introduce these two corpuses in detail.

4.1.1 QuoteLi3

QuoteLi3 (Muzny, Fang, Chang, & Jurafsky, 2017) is a comprehensive dataset of literary narratives. It contains three novels which are Jane Austen's *Pride and Prejudice*, *Emma*, and Anton Chekhov's *The Steppe*. This was built based on two corpuses: the Columbia Quoted Speech Corpus (Elson & McKeown, 2010) and the data provided by He, Barbosa, and Kondrak (2013).

The Columbia Quoted Speech Corpus (CQSC) is a large dataset in the domain of literary narratives. This corpus includes both quote-mention and

quote-speaker labels (Elson & McKeown, 2010). It uses Stanford NER tagger (Finkel, Grenager, & Manning, 2005) and a method stated by Davis, Elson, and Klavans (2003) to detect the possible speakers. However, there are some shortcomings associated with the use of low accuracy tools and crowdsourced labels. The quote-mention labels in this corpus are labeled by 3 different annotators, after the annotation only 65% of the quotes in the CQSC had unanimous agreement. However, 17.6% of the quotes in this corpus were unlabeled (Elson & McKeown, 2010). The quote-speaker label of this corpus is generated based on the quote-mention label by an off the shelf coreference tool. However, due to the limited performance of the coreference tool, the quality of the quote-speaker labels are not very reliable. Muzny et al. (2017) stated that 48.1% of the quotes do not have a clear speaker label and 9.7% of speakers can not be associated to a known character entity (Muzny et al., 2017). Another research studied on this corpus also found that 8% of quotations in this corpus are labeled incorrectly (O'Keefe et al., 2012).

Another dataset used to build the QuoteLi3 corpus is the data from He et al. (2013). This corpus does not have quote-mention labels but the quote-speaker labels in it are of a high quality. In this dataset, all of the quoted texts which are in the same paragraph are assumed to be attributed to one

speaker. Generally, there is only one speaker in each paragraph in first-level quotes. However, sometimes this assumption could be broken down. For example, in nested quotes, the speaker of the quotes are rarely to be the same one. This assumption can't be correct in the books which include more complex conversational structures, such as *The Steppe*. While it is correct for *Pride and Prejudice*.

The data provided by He et al. (2013) and CQSC does not have any overlap. In QuoteLi3, the unlabeled data from both datasets are annotated, and some incorrect data is corrected. For both quote-mention and quote-speaker labels, this dataset provides a complete set of annotations. It has 3103 individual quotes from three novels, and each quote linked to a speaker and mentioned label. The total number of labels in this corpus is 6206, and more than 3000 of these labels are newly annotated. The author of the QuoteLi3 corpus also provides a browser-based annotation tool and presented in detail how they annotated the quotations in the corpus.

4.1.2 Reuters Corpus

The Reuters Corpus data (RCV1) (Lewis, Yang, Rose, & Li, 2004) is used for the cross domain test of the entire system. This dataset was provided by

Reuters (Read, 1999), and it contains more than 800,000 news articles. According to the theme of these news articles, they have been manually divided into four groups: Corporate or Industrial data (CCAT), Economics data (ECAT), Government or Social data (GCAT), and Markets data (MCAT). The data used in this thesis is randomly selected from the Corporate or Industrial data.

This corpus does not have labels for the author identification task. Although this corpus will not be used for training the machine learning models, this data still needs to be labeled to test the generalisation performance of the models.

Data Annotation

A simple programme written in Python is used to assist with manual data labeling. The programme automatically calculates the location of the tag data in the context and saves the record to the text. About 100 files are labeled, several of which are as follows.

{'speaker_id': '1103', 'start_index': 6, 'end_index': 25}

{'speaker_id': '1103', 'start_index': 0, 'end_index': 13}

These examples are the labeled data of these two following sentences:

The New York Stock Exchange said Tuesday that a seat on the exchange sold for $1.16 million, down $287,500 from the previous sale on May 7.

The current level of bids was $1 million and offers were at $1.25 million, it said.

The names of the speakers are stored in another file, which can effectively reduce the surplus of data. This is because a speaker may correspond to multiple quotations, just like the one shown in the example. For quotation extraction, the content of the target quotation is not important. As long as the start index and the end index is matched with the label, the model extracts the correct data. The total amount of annotated quotation is 248 quotations from 100 files.

4.2 Pre-processing of Dataset

Natural Language Toolkit (NLTK) (Bird & Loper, 2004) is used in the data pre-processing step. It is a set of natural language processing tools based on the programming language Python (Steven, Klein, & Loper, 2009). The raw data is pre-processed in the following steps.

Tokenization : This is a process of breaking a stream of text up into

tokens. The NLTK Library has the word_tokenize and sent_tokenize to easily break a stream of text into a list of words or sentences respectively. For further processing, the inputs will be the list of tokens rather than the original texts.

Part-of-Speech : This is the basic step to mark the part of each word. We used the pre-trained HMM model provided by NLTK for POS tagging.

Name Entity Recognition (NER) : This is used to extract all the possible speakers from the texts. The data in the QuoteLi3 corpus is stored in xml format. In this corpus, all the possible speakers are labeled out, and there is no need to use an NER tagger to detect all the named entities. The character lists in the corpus are extracted as the speakers. However, in the RCV1 corpus and real world data there is no character list; for the generalisation purpose, the NER is necessary for identifying the potential speakers. This text is processed using an HMM model and chunks of consecutive proper nouns are extracted. The chunks with the "Location" tag are excluded because it is not possible that they are the speakers.

Dependency parsing : As the indirect and mixed quotations usually appear as a clause, the structure of the sentence should be parsed. The Stanford dependency parser (Chen & Manning, 2014) is used to retrieve

both the sentence dependencies and the phrase structure parsing.

Word Lemmatization: The purpose of this step is to reduce the inflectional forms of each verb into a common base. Word Lemmatization could make the steps of keyword matching more efficient and reduce the number in the speech verb lexicon. Depending on the part of speech, word lemmatization can discriminate between words which have different meanings with the knowledge of the context around the target speech verb.

4.3 Experimental Environment

The experiment is run on a laptop with Intel Core i7 CPU 2.50GHZ. The Operating System installed on the laptop is 64-bit Microsoft Window 10. The project of author identification in free text is developed in Python programming language, and the Integrated Development Environment (IDE) used during the development and implementation is JetBrains PyCharm Professional Edition 2018.1.2.

4.4 Extraction Result

The test set used for the extraction test is from the novel *Emma*. QuoteLi3 corpus provides a well annotated data for the first 21 chapters of the novel *Emma*. The test set has about 20% of the total number of quotation from the novel *Emma* in QuoteLi3. The total number of quotations used in the testing is 742. This rule based model achieved a precision value of 1.00, which means that all the sentences extracted by the model are quotations. However, the recall of this model is 0.846. The model extracted 628 quotations from the test set while the total number of quotations in the novel *Emma* is 742; thus the F-measure of this quotation extraction model is 0.88.

4.5 Attribution Results

The test set used for this evaluation is about 20% of the total number of quotations extracted from the novel *Pride and Prejudice*. There are 315 quotations randomly extracted from this novel as the test set. In the novel *Pride and Prejudice*, there are 32 possible speakers. Thus, the total number of quote-speaker pairs in the test set is 10080. Here are the results of the test set, with the CRF model and the SVM model.

4.5.1 CRF Model Result

The predicted result of binary CRF model only is presented in the following Table 4.1.

	Actual Positive	***Actual Negative***
Predicted Positive	218	42
Predicted Negative	97	9723

Table 4.1: Results of the Binary CRF model.

From Table 4.1, the accuracy of the binary model can be calculated, which is 98.6%. However, the accuracy is not reliable due to most of the data in the test set having a negative label. Even in the model predicted, if all the examples are negative, it can still achieve 96.875% accuracy. In this situation, precision, recall and F-score can show more reliable information about the performance of the model.

The numbers in Table 4.1 were used to calculate the precision, recall and F-score by the formulas introduced in the previous chapter. The precision of the model is 83.8% and recall is 69.2%. The F-score of the binary model is

75.8%.

For each quotation, one final result will be selected from the 32 quote-speaker pairs . After reconciling these results into a single decision for each quote, 264 quotations are correctly attributed to their authors. The CRF model produces 83.8% accuracy.

4.5.2 SVM Model Result

The SVM model used the same features as the CRF model. The main difference between these two models is that the CRF model does the binary classification first and the SVM model directly selects the best candidate from the 32 quote-speaker pairs for each quotation. The test set used for these two models is also the same one. In the evaluation results of this model, there are 270 quotations which are attributed to their authors correctly. While the total number of quotations in the test set is 315, the accuracy of the SVM model is 85.7%.

4.6 Overall Result

The aim of the overall test is to test the entire system with both the rule based approach for quotation extraction and the machine learning model for speaker attribution. The input data is raw data. The xml file is unlabeled and stored in a text file.

There should be 1575 quotations in the test set. After the quotation extraction step there are 1363 quotations which are extracted from the test data and all of these quotations are extracted correctly. For the speaker attribution part, with the CRF model, 1202 quotations are attributed correctly. By switching to the SVM model, the result of the correctly attributed quotation is 1227. Table 4.2 shows the accuracy of these two models.

Method	***Accuracy***
System with CRF Model	76.3%
System with SVM Rank Model	77.9%

Table 4.2: The overall results of the two models on the test set.

4.7 Generalisation Result

The main challenge of cross domain testing is that the models may have data sparsity and coverage issues in functionality. These are issues mainly because the training and testing are in different domains. This set of experiments uses the QuoteLi3 data set as the training corpus used for models is the QuoteLi3 corpus, but the testing set used in this generalisation test is from the RCV1 corpus. Thus the system is not able to perform as well as the test with the same domain data.

In generalisation testing, it is a correct prediction as long as the quotation extraction model can detect more than half of the content in the quotation and is attributed to the correct speaker by the speaker attribution model. Table 4.3 shows the results for both the CRF model and the SVM model on the RCV1 corpus:

The accuracy drops by roughly 4% when the model is applied to the news domain data. The accuracy of the SVM model is still higher than the CRF model.

Method	***Accuracy***
System with CRF Model	72.3%
System with SVM Rank Model	73.6%

Table 4.3: Results of two models on the generalisation test set.

4.8 Limitations of the Experiment

The number of quote-speaker pairs increase, based on the number of possible speakers and quotations there are in the document. The amount of generalisation test data is not enough. This experiment did not consider the information of the quote itself. For example, assuming that the same speaker tends to talk about similar topics, the quotations can be measured by content similarity.

Chapter 5

Analysis and Discussion

This chapter will present the discussion and result analysis according to the outcomes of the experiment. The errors and the reasons for them are analysed in this approach; this could improve the approach for future work. The comparison of two models for speaker attribution will also be discussed. There is also a discussion for the reason for a drop in accuracy in the generalisation test.

5.1 Error Analysis

This section presents the analysis of the main sources of error for the author identification system. The error analysis contain two main parts, extraction

error from the rule based approach and the attribution error from the machine learning models. The examples used in this error analysis are from False Negative and False Positive results in the experiment.

5.1.1 Extraction Error

The main sources of error for the quotation extraction system can be divided into three types: nested quotations, unclear boundary and semantic ambiguity.

Nested Quotations : Nested quotations exist in another direct quotation, and usually do not have a clear cue. The content of quotations is not focused in this system, thus the nested quotations instead the first-level quotation may be missed. This kind of error could lead to a low recall of the quotation extraction system. Here is an example from the novel *Emma*:

"But I am afraid, Mr. Elton, Harriet will not like to sit. She thinks so little of her own beauty. Did not you observe her manner of answering me? How completely it meant, ***'why should my picture be drawn?'****"*

In this example, the nested quotation in the first-level quotation is unextracted. While there is no speech verb near the quotation, the rule based

approach makes the wrong decision.

Unclear Boundary : This kind of error often occurs in the extraction of indirect and mixed quotations. On the generalisation test, the recall of the rule based approach does not change much but the precision drop a lot due to the incomplete recognition of indirect and mixed quotations. In some cases, even for human annotators, the boundaries of quotations are ambiguous. Determining whether a sentence is still part of the quotation content usually requires semantic understanding and world knowledge. Here are two examples from the Reuters Corpus:

*Merck and Co said **Tuesday** it sold its stake in Ostex International Inc or 736,844 common shares.*

*Derma Sciences Inc said **on Tuesday that** the Republic of the Philippines has issued provisional Certificates of Product Registration for its line of wound care products.*

In these two examples, the word 'Tuesday' should not be part of the quotation content.

Semantic ambiguity: This kind of error usually due to the cue of a quotation is unclear, sometimes the speech verb may not be one verb but a

phrase. Here is an example from the Reuters Corpus:

Federal aviation authorities on Tuesday ended a delay in issuing new anti-terrorism rules for cargo shippers, ***sending out a notice*** *that the regulations would now go into effect on August 28.*

In the example, the content after 'sending out a notice' should be extracted as a quotation.

5.1.2 Attribution Error

This section presents an overview of the most common sources of error affecting the model identifying the source span. The main sources of error for the speaker attribution models contain two types: incorrect entire or mention and parse error.

Incorrect entity or mention: This kind of error usually due to the error from NER tagger. The wrong entity was identified, causing the span of speaker to be incorrect. This error may also break down the structure of the sentence. Here is an example from the Reuters Corpus:

"Our net growth has increased dramatically ... All the key drivers of the business, we are succeeding on," ***Managing Director Hans Snook*** *said*

in an interview.

In this example, the NER tagger chunks 'Managing Director' and 'Hans Snook' as two different person, this error makes the speaker attribution system attribute the wrong speaker to the quotation.

Parse error: This kind of error usually occurs when the sentence structure is complicated. Especially for those sentences in which the speaker is far away from the speech verb and there are other pronouns or names between them. Here is an example from the novel *Emma*:

he *might constrain himself, while the ladies were comfortably clearing the nicer things, to say:*

In this example, the pronoun 'he' should be the mention to quote while the model attribute 'the ladies' as the mention to quote due to the lower number in the distance feature.

5.2 Model Comparison and Discussion

It can be seen from the above results that the performance of the SVM model is slightly better than the CRF model. This is because the CRF model judges each pair of data separately, without considering the relationship between

other speakers and the data. Although the candidate for the same sentence was selected and compared to the best option at the end, this effect was not considered in the process of model judgment and training. The SVM model compares each pair of data, and the relationship between other candidates and sentences is considered in the training process.

The research on other areas such as chemical entity recognition (Tang et al., 2015) it is also shown that the model build with the SVM performs better than the CRF model. This phenomenon occurs mainly because the SVM-based system has a higher recall rate when using the same features in NER tasks. Another research in a similar field (D. Li, Savova, & Kipper-Schuler, 2008) shows that in their experiment, the performances of the CRF model is better than the SVM model. This research stated that without Bag of Word (BOW) features, the SVM model outperformed the CRFs. Where the BOW feature has been added, then the performance of the CRF model has been significantly improved. The precision and recall of the CRF model are both higher than the SVM model.

Therefore, both models have their own merits, and choosing the right model in different situations can achieve better performances. In this thesis, the SVM model is more suitable for the speaker attribution task.

5.3 Generalisation Performance Discussion

Compared to the overall test on the QuoteLi3 Corpus, the results of the generalisation test on the Reuters set dropped about 4%. In the QuoteLi3 Corpus, most quotations are direct quotation, this reduces the challenge for a rule based approach to extract the correct quotations. However, in the news domain such as data in the Reuters Corpus, the number of indirect and mixed quotations are more than in literary narratives. Although the precision of the quotation extraction dropped a lot, the incomplete quotations extracted by the rule based approach can also be attributed to the correct speakers. So the overall accuracy of the entire system only dropped 4% from the literary narrative to the news domain.

5.4 Comparison with Other Research

Compared to the work of Pareti et al. (2013), the rule based approach used in this thesis has 8% higher precision than the supervised learning approach in that research. However, the recall of the rule based approach is slightly lower than the supervised learning approach. The use of the supervised learning method in the quotation extraction task can slightly improve the recall of

the system but the precision will drop a lot. Elson and McKeown (2010) also use a rule based approach to extract quotations, with supervised learning methods being used for quote attribution. In their research, they use logistic regression to attribute the quotations, while in this thesis the CRF model and SVM model are used for quotation attribution. The overall accuracy of Elson and McKeown (2010) is 83%. Our approach get 5% lower than their system.

Chapter 6

Conclusion

In this chapter, a conclusion of this thesis will be presented. By the end of this thesis, possible future work resulting from this research will be pointed out.

6.1 Summary

This thesis examines the author identification task in free text. As part of this work, the linguistic structure of quotations in the text is first analysed. The author identification system is divided into two parts: the quotation extraction part and speaker attribution part. A rule based approach is then developed for quotation extraction, and machine learning models are used

for identifying speaker attribution. The machine learning models include two different models: a binary model with a reconciliation method and a multi-classification model. A cross domain test is taken to evaluate the generalisation performance of the system. This work provides an error analysis of both linguistic and systematic errors. After the study and experiment, all the research questions discussed in Chapter 1 have solutions.

6.2 Future Work

The objective of this thesis is to do the author identification in free text as mentioned in Chapter 1. However, only one language is considered in the experiment. In further investigations, it is possible to identify more languages for author identification tasks. The field of resources used in the experiment involves only literary narratives and news. Future work could include a wider range of resources. Other kinds of machine learning methods are also available for this task. Annotated data resources are always expensive to obtain; future research can be more about using semi-supervised learning and unsupervised learning to accomplish the author identification task.

The results of this study can be used in the field of social network ex-

traction for novels. In the text to speech area, the approach used for author identification can also help to build high quality audios. By analysing all the quotations spoken by one person, it can also help to determine the personality of that person. In the news domain, such as in political news, the opinion of the speakers can also be found out through the content of their speech.

Bibliography

Andrew, G., & Gao, J. (2007). Scalable training of l1-regularized log-linear models. In *International Conference on Machine Learning* (International Conference on Machine Learning). Retrieved from https://www.microsoft.com/en-us/research/publication/scalable-training-of-l1-regularized-log-linear-models/

Atkinson, J., Heritage, J., & Oatley, K. (1984). *Structures of social action*. Studies in Emotion and Social Interaction. Cambridge University Press. Retrieved from https://trove.nla.gov.au/work/22229804?selectedversion=NBD3225919

Bird, S., & Loper, E. (2004). NLTK: the natural language toolkit. In *Proceedings of the ACL 2004 on Interactive poster and demonstration sessions* (p. 31). Association for Computational Linguistics.

Celikyilmaz, A., Hakkani-Tur, D., He, H., Kondrak, G., & Barbosa, D. (2010). The actortopic model for extracting social networks in literary narrative. In *NIPS Workshop: Machine Learning for Social Computing*. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.226.7481

Chapple, E. D. (1940). "Personality" differences as described by invariant properties of individuals in interaction. *Proceedings of the National Academy of Sciences of the United States of America*, *26*(1), 10.

Chen, D., & Manning, C. (2014). A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 740–750).

Christian Nørgaard Storm Pedersen, T. M. (2012). Hidden Markov models terminology and basic algorithms. Retrieved June 11, 2018, from http://www.cs.au.dk/~cstorm/courses/PRiB/slides/hidden-markov-models-1.pdf

Davis, P. T., Elson, D. K., & Klavans, J. L. (2003). Methods for precise named entity matching in digital collections. In *2003 Joint Conference on Digital Libraries, 2003. Proceedings.* (pp. 125–127). IEEE.

de La Clergerie, É., Sagot, B., Stern, R., Denis, P., Recourcé, G., & Mignot, V. (2009). Extracting and visualizing quotations from news wires. In *Language and Technology Conference* (pp. 522–532). Springer.

Dekker, N., Kuhn, T., & van Erp, M. (2018). Evaluating social network extraction for classic and modern fiction literature. *PeerJ Preprints*, *6*, e27263v1.

Eddy, S. R. (1996). Hidden Markov models. *Current Opinion in Structural Biology*, *6*(3), 361–365.

Elson, D. K., Dames, N., & McKeown, K. R. (2010). Extracting social networks from literary fiction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 138–147). Association for Computational Linguistics.

Elson, D. K., & McKeown, K. R. (2010). Automatic attribution of quoted speech in literary narrative. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*.

Finkel, J. R., Grenager, T., & Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Com-*

putational Linguistics (pp. 363–370). Association for Computational Linguistics.

Glass, K., & Bangay, S. (2006). Hierarchical rule generalisation for speaker identification in fiction books. In *Proceedings of the 2006 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on IT Research in Developing Countries* (pp. 31–40). South African Institute for Computer Scientists and Information Technologists.

Glass, K., & Bangay, S. (2007). A naive salience-based method for speaker identification in fiction books. In *Proceedings of the 18th Annual Symposium of the Pattern Recognition Association of South Africa (PRASA'07)* (pp. 1–6).

He, H. (2011). Automatic speaker identification in novels. Retrieved from era.library.ualberta.ca

He, H., Barbosa, D., & Kondrak, G. (2013). Identification of speakers in novels. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Vol. 1, pp. 1312–1320).

Herbrich, R. (2000). Large margin rank boundaries for ordinal regression. *Advances in large margin classifiers*, 115–132.

Hsu, C.-W., & Lin, C.-J. (2002). A comparison of methods for multiclass support vector machines. *IEEE transactions on Neural Networks*, *13*(2), 415–425.

Iosif, E., & Mishra, T. (2014). From speaker identification to affective analysis: A multi-step system for analyzing children's stories. In *Proceedings of the 3rd Workshop on Computational Linguistics for Literature (CLFL)* (pp. 40–49).

Joachims, T. (2002). Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 133–142). ACM.

Joachims, T. (2006). Training linear SVMs in linear time. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 217–226). ACM.

Krestel, R., Bergler, S., Witte, R., et al. (2008). Minding the source: Automatic tagging of reported speech in newspaper articles. *Reporter*, *1*(5), 4.

Kudo, T. (2005). Crf++: Yet another crf toolkit. *http://crfpp. sourceforge. net/*.

Lewis, D. D., Yang, Y., Rose, T. G., & Li, F. (2004). RCV1: A new benchmark collection for text categorization research. *Journal of Machine Learning Research*, *5*(Apr), 361–397.

Li, D., Savova, G., & Kipper-Schuler, K. (2008). Conditional random fields and support vector machines for disorder named entity recognition in clinical texts. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing* (pp. 94–95).

Li, H. (2011). A short introduction to learning to rank. *IEICE TRANSACTIONS on Information and Systems*, *94*(10), 1854–1862.

Liang, J., Dhillon, N., & Koperski, K. (2010). A large-scale system for annotating and querying quotations in news feeds. In *Proceedings of the 3rd International Semantic Search Workshop* (p. 7). ACM.

Liu, T.-Y. et al. (2009). Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, *3*(3), 225–331.

Mairesse, F., & Walker, M. (2006). Automatic recognition of personality in conversation. In *Proceedings of the Human Language Technology Con-*

ference of the NAACL, Companion Volume: Short Papers (pp. 85–88). Association for Computational Linguistics.

Mairesse, F., Walker, M. A., Mehl, M. R., & Moore, R. K. (2007). Using linguistic cues for the automatic recognition of personality in conversation and text. *Journal of Artificial Intelligence Research*, *30*, 457–500.

Malouf, R. (2002). A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of the 6th Conference on Natural Language Learning-Volume 20* (pp. 1–7). Association for Computational Linguistics.

Mamede, N., & Chaleira, P. (2004). Character identification in children stories. In *International Conference on Natural Language Processing (in Spain)* (pp. 82–90). Springer.

Matteucci, M. (2008). Three fundamental questions with Hidden Markov Models. Retrieved June 11, 2018, from http://home.deib.polimi.it/matteucc/MSI/download/FundamentalIssuesHMM.pdf

Mejer, A., & Crammer, K. (2010). Confidence in structured-prediction using confidence-weighted models. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing* (pp. 971–981). Association for Computational Linguistics.

Mitchell, T. M. et al. (1997). Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, *45*(37), 870–877.

Muhuri, S., Chakraborty, S., & Chakraborty, S. N. (2018). Extracting Social Network and Character Categorization From Bengali Literature. *IEEE Transactions on Computational Social Systems*, *5*(2), 371–381.

Muzny, G., Fang, M., Chang, A., & Jurafsky, D. (2017). A two-stage sieve approach for quote attribution. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers* (Vol. 1, pp. 460–470).

Ng, A. Y., & Jordan, M. I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In *Advances in Neural Information Processing Systems* (pp. 841–848).

Nocedal, J. (1980). Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, *35*(151), 773–782.

O'Keefe, T., Pareti, S., Curran, J. R., Koprinska, I., & Honnibal, M. (2012). A sequence labelling approach to quote attribution. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 790–799). Association for Computational Linguistics.

Okazaki, N. (2007). CRFsuite: a fast implementation of Conditional Random Fields (CRFs). Retrieved from http://www.chokkan.org/software/crfsuite/

Pareti, S., O'Keefe, T., Konstas, I., Curran, J. R., & Koprinska, I. (2013). Automatically detecting and attributing indirect quotations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing* (pp. 989–999).

Pavllo, D., Piccardi, T., & West, R. (2018). Quootstrap: Scalable unsupervised extraction of quotation-speaker pairs from large news corpora via bootstrapping. In *Twelfth International AAAI Conference on Web and Social Media*.

Pouliquen, B., Steinberger, R., & Best, C. (2007). Automatic detection of quotations in multilingual news. In *Proceedings of Recent Advances in Natural Language Processing* (pp. 487–492).

Qi, L., & Chen, L. (2010). A linear-chain crf-based learning approach for web opinion mining. In *International Conference on Web Information Systems Engineering* (pp. 128–141). Springer.

Ramsay, R. W. [Ronald W]. (1968). Speech patterns and personality. *Language and Speech*, *11*(1), 54–63.

Ramsay, R. (1966). Personality and speech. *Journal of Personality and Social Psychology*, *4*(1), 116.

Read, D. (1999). *The power of news: the history of Reuters*. Oxford University Press, USA.

Rosen-Zvi, M., Griffiths, T., Steyvers, M., & Smyth, P. (2004). The author-topic model for authors and documents. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence* (pp. 487–494). AUAI Press.

Sagot, B., Danlos, L., & Stern, R. (2010). A lexicon of French quotation verbs for automatic quotation extraction. In *7th International Conference on Language Resources and Evaluation-LREC 2010*.

Salamin, H., Favre, S., & Vinciarelli, A. (2009). Automatic role recognition in multiparty recordings: Using social affiliation networks for feature extraction. *IEEE Transactions on Multimedia*, *11*(7), 1373–1380.

Salamin, H., Vinciarelli, A., Truong, K., & Mohammadi, G. (2010). Automatic role recognition based on conversational and prosodic behaviour. In *Proceedings of the 18th ACM International Conference on Multimedia* (pp. 847–850). ACM.

Salway, A., Meurer, P., Hofland, K., & Reigem, Ø. (2017). Quote extraction and attribution from Norwegian newspapers. In *Proceedings of the 21st Nordic Conference on Computational Linguistics* (pp. 293–297).

Sarmento, L., & Nunes, S. (2009). Automatic extraction of quotes and topics from news feeds. In *DSIE'09-4th Doctoral Symposium on Informatics Engineering*.

Scheible, C., Klinger, R., & Padó, S. (2016). Model architectures for quotation detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (Vol. 1, pp. 1736–1745).

Schneider, N., Hwa, R., Gianfortoni, P., Das, D., Heilman, M., Black, A., et al. (2010). *Visualizing topical quotations over time to understand news discourse*. Technical Report, TR CMU-LTI-10-013, Carnegie Mellon University, Pittsburgh (2010).

Seymore, K., McCallum, A., & Rosenfeld, R. (1999). Learning hidden Markov model structure for information extraction. In *AAAI-99 Workshop on Machine Learning for Information Extraction* (pp. 37–42).

Shalev-Shwartz, S., Singer, Y., Srebro, N., & Cotter, A. (2011). Pegasos: Primal estimated sub-gradient solver for SVM. *Mathematical Programming*, *127*(1), 3–30.

Smeros, P., Castillo, C., & Aberer, K. (2019). Scilens: Evaluating the quality of scientific news articles using social media and scientific literature indicators. *arXiv preprint arXiv:1903.05538*.

Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, *45*(4), 427–437.

Steven, B., Klein, E., & Loper, E. (2009). Natural language processing with Python. *OReilly Media Inc*.

Syaifudin, Y., & Nurwidyantoro, A. (2016). Quotations identification from Indonesian online news using rule-based method. In *2016 International Seminar on Intelligent Technology and Its Applications (ISITIA)* (pp. 187–194). IEEE.

Tang, B., Feng, Y., Wang, X., Wu, Y., Zhang, Y., Jiang, M., et al. (2015). A comparison of conditional random fields and structured support vector machines for chemical entity recognition in biomedical literature. *Journal of Cheminformatics*, *7*(1), S8.

Taylor, H. C. (1934). Social agreement on personality traits as judged from speech. *The Journal of Social Psychology*, *5*(2), 244–248.

Tsochantaridis, I., Joachims, T., Hofmann, T., & Altun, Y. (2005). Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, *6*(Sep), 1453–1484.

Vapnik, V. (1999). *The nature of statistical learning theory*. Springer Science & Business Media.

Yeung, C. Y., & Lee, J. (2017). Identifying speakers and listeners of quoted speech in literary works. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)* (pp. 325–329).

Zhang, J. Y., Black, A. W., & Sproat, R. (2003). Identifying speakers in children's stories for speech synthesis. In *Eighth European Conference on Speech Communication and Technology*.