A SENTIMENT BASED AUTOMATIC QUESTION-ANSWERING FRAMEWORK

by

Qiaofei Ye

A Thesis

Submitted to the Faculty of Purdue University

In Partial Fulfillment of the Requirements for the degree of

Master of Science

Department of Computer & Information Technology

West Lafayette, Indiana

May 2019

THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL

Dr. Julia Rayz, Chair

Department of Computer and Information Technology

Dr. Baijian Yang, Member

Department of Computer and Information Technology

Dr. Ida Ngambeki, Member

Department of Computer and Information Technology

Approved by:

Dr. Eric T Matson

Head of the Graduate Program

I dedicate my thesis to my parents and friends, who supported me and helped me through my Master study. Special thanks to my advisor Julia Rayz, guides me through the field of NLP, inspires me with rigorous logical flow, and being our good mentor and friend.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory Cell
biLSTM	Bidirectional Long Short-Term Memory Cell
BRNNs	Bidirectional Recurrent Neural Nets
QA	Question-Answering
Non-Factoid QA	Non-factoid Question-Answering

ABSTRACT

Author: Ye, Qiaofei,. MS
Institution: Purdue University
Degree Received: May 2019
Title: A Sentiment Based Automatic Question-Answering Framework
Committee Chair: Julia Rayz

With the rapid growth and maturity of Question-Answering (QA) domain, non-factoid Question-Answering tasks are in high demand. However, existing Question-Answering systems are either fact-based, or highly keyword related and hard-coded. Moreover, if QA is to become more personable, sentiment of the question and answer should be taken into account. However, there is not much research done in the field of non-factoid Question-Answering systems based on sentiment analysis, that would enable a system to retrieve answers in a more emotionally intelligent way. This study investigates to what extent could prediction of the best answer be improved by adding an extended representation of sentiment information into non-factoid Question-Answering.

Keywords: Non-factoid Question-Answering, Sentiment analysis, Long Short-Term Memory

CHAPTER 1. INTRODUCTION

1.1 Introduction

With the rapid growth of internet, non-factoid Question-Answering tasks (open-ended Question-Answering) are in high demand. For example, virtual personal assistants and Question-Answering bots on E-commerce websites. In real-world applications, people often encounter situations where people come up with a question but only get answers that do not match with his/her expectations. The reason behind is that, the open-ended questions, such as why-question or how-question, often follow the pattern that "if something undesirable happens, the reason is also often something undesirable, and if something desirable happens, the reason is also often something desirable." (Oh et al., 2012). Thus, if the answer contains the wrong sentiment or something that is not as desirable as the question, the user will regard it as an answer that is not that appropriate. In other words, even when the knowledge contained in the answer is correct, if the sentiment contained in the answer is not considered empathetic, this Question-Answering interaction is not considered emotionally intelligent.

Addressing the same problem, Mishra et al. (2016) stated that WHY questions asked in Opinion Question-Answering systems (OQAS) "expect answers to incorporate reasons and explanations for the questioners' sentiment expressed in the questions" (Mishra et al., 2016). For this reason, when the Question-Answering System is designed, not only the quality of the fact or information contained in the answer should be considered, but also how the sentiment is addressed. Thus, when having several candidate answers in the answer pool that belong to the same question, knowing how to choose the answer that both indicates correct information and contains proper sentiment may be of great importance in non-factoid Question-Answering tasks.

The goal of this research is to improve Question-Answering framework to retrieve the best answer in a more emotionally intelligent way. More specifically, in non-factoid Question-Answering domain, this study investigates whether adding sentiment into non-factoid Question-Answering can help improve the performance of retrieving the best answer.

1.2 Statement of the Problem

Non-factoid Question-Answering tasks are quite popular today. However, these Question-Answering tasks are either pure information-based, which means the sentiment contained in the question does not affect which answer is chosen as the best answer, or highly keyword related and hard-coded. So far, there is not much research done in the field of non-factoid Question-Answering systems based on sentiment analysis to retrieve answer in a more emotionally intelligent way.

This research explores a new method using LSTM (Long Short-Term memory) and combining sentiment information to select the best answers from the questions pool automatically. The method takes the sentiment factor in the question and the candidate answers into account. The best answer is selected by combining Natural language features, and also sentiment information.

From previous research, we can conclude that when users ask questions, they also expect answers with the same sentiment polarity as the questions (Mishra et al., 2016). However, to the best of our knowledge, previous research that considers sentiment information into the QA system is all based on sentiment polarity, for example, ‘positive’ vs. ‘negative’. An extended representation of sentiment has not been considered. Thus, after getting an extended representation of sentiment in questions and answers (Felbo et al., 2017), it is crucial to verify whether adding extended sentiment information into non-factoid Question-Answering could improve best answer prediction performance.

Additionally, previous non-factoid QA systems with sentiment information seldom use English corpus (Oh et al., 2012), and are based on “Why” questions (Oh et al., 2012; Mishra et al., 2016). Performing this task on “How” questions has never been conducted. This research concentrates on the “How” question.

1.3 Research Question

The research question is: in non-factoid Question-Answering based on biLSTM in English, to what extent could prediction of the best answer be improved by considering an extended representation of sentiment information of questions and answers?

1.4 Hypothesis

The hypothesis is: in non-factoid Question-Answering in English, there is a correlation between sentiment information of question/answer and best answer, as selected by users.

1.5 Scope

The aim of the study is to add sentiment information into non-factoid Question-Answering system, based on "How" questions, with the goal of improving the best answer prediction ability of the system. This would involve finding a way to compute and represent the sentiment information in the questions/answers, and also incorporating the sentiment information into an existing QA neural network for answer prediction. For the neural network, existing implementations of encoding networks used in Question-Answering tasks are reused. The majority of the study will be focusing on adding and incorporating the sentiment information to the neural network, and make the neural network attend to it. The performance of the network is evaluated using the same metrics which are used in the baseline (Tan et al., 2016).

1.6 Significance

With the rapid growth of the internet, a majority of the existing "Question-Answering (QA) systems serve the needs of answering factual questions such as "When was James Dean born?" and "Who won the Nobel Peace Prize in 1991?". In addition to facts, people would also like to know about others' opinions, thoughts, and feelings toward some specific topics, groups, and events" (Oh et al., 2012). While previous non-factoid QA research only focuses on "Why" questions in English, this work is applied to "How" question in non-factoid Question-Answering.

Adding sentiment analysis to non-factoid Question-Answering tasks has rarely been conducted. Previous work adopted manually labeled data for sentiment/opinion information (Stoyanov et al.,

2005; Somasundaran et al., 2007), or generated sentiment information in a simple form of polarity score (Oh et al., 2012). This research adopted an extended representation of sentiment, and combined with the non-factoid QA network to find out whether adding sentiment information could lead to an improvement in Question-Answering task performance. To the best of our knowledge, this is also the first work combining sentiment information into non-factoid QA in English for "How" questions.

This research will expand traditional sentiment computation/representation with a more precise sentiment representation, from binary (positive/negative) or ternary positive/negative/neutral) or single scale (-4 to +4), to an extended scale of a more diverse set of noisy labels of 64 dimensions. The sentiment information thus has richer representations than that of sentiment polarity. With this extended sentiment representation, evaluations will be performed to validate how sentiment score affects the performance of the non-factoid Question-Answering system.

1.7 Assumptions

The following assumptions are made for this study:

- The performance metrics used for evaluation are assumed to accurately convey the quality of the Question-Answering model.
- The sentiment computation method is able to generate sentiment information that could represent the sentiment contained in the questions and answers with the same judging criterion.

1.8 Definitions and Main concepts

Non-factoid Question-Answering (QA): Non-factoid Question-Answering represents all Question-Answering topics beyond factoid Question-Answering, while a factoid QA is about providing concise facts. For example, "who is the headmaster of Hogwarts?", "What is the population of Mars". In contrast, a non-factoid question can be about anything. The QA system can expect question asking about an answer to a math problem, to explain how to fix a specific model of a car, and so on, so forth. Answering multiple-choice questions also belong to the area of non-factoid QA, there might be some overlap with factoid QA in this task (Yang et al., 2016).

Sentiment Polarity: “Semantic orientation, or polarity, is a consistent lexical property with a high inter-rater agreement” (Hatzivassiloglou et al.,1997).

Recurrent Neural Network (RNN): “Recurrent networks can, in principle, use their feedback connections to store representations of recent input events in the form of activations. The most widely used algorithms for learning what to put in short-term memory” (Hochreiter et al., 2001).

Long Short-term memory neural nets (LSTM): “The Long Short Term Memory architecture (Hochreiter and Schmidhuber, 1997, Gers et al., 2002) was motivated by an analysis of error flow in existing RNNs (Hochreiter et al., 2001), which found that long time lags were inaccessible to existing architectures, because backpropagated error either blows up or decays exponentially” (Hochreiter and Schmidhuber, 1997, Gers et al., 2002).

Bidirectional recurrent neural nets (BRNNs): “The basic idea of bidirectional recurrent neural nets (BRNNs) (Schuster and Paliwal, 1997, Baldi et al., 1999) is to present each training sequence forward and backward to two separate recurrent nets, both of which are connected to the same output layer” (Graves et al., 2005).

1.9 Limitations

The study is undertaken with the following limitations:

- The question-answer pairs the model is trained on, are limited to “How” questions.
- For this study, there is one and only one best answer to each question.

1.10 Delimitations

The study acknowledges the following delimitations:

- This study only focusses on the sentiment information based on the chosen sentiment framework, only consider sentiment dimensions generated by the chosen sentiment computation framework.

- There are more sentiment analysis frameworks available. However, they are not being selected for this study, since they only offer distant labels of two or more categories like ‘positive’, ‘neutral’, ‘sentiment’.
- There might be a dataset with higher subjective questions and questions with sentiment ratio. However, this study is restricted to use the Yahoo! Answers Manner Questions dataset.

CHAPTER 2. LITERATURE REVIEW

This chapter is a summary of the recent research in Question-Answering, Answer Ranking tasks, and Sentiment Analysis.

2.1 Introduction

Non-factoid Question-Answering task is one of the fundamental problems in Natural Language Processing tasks. Compared to factoid QA, non-factoid QA tasks do not have a specific 'correct' answer to each question. That is the reason why non-factoid Question-Answering applies to open-answer tasks.

For the following sections, existing Question-Answering systems are described and compared. Then, since this study is specifically about answer ranking, the review of current Question-Answering framework with answer ranking and answer selection is covered as well. With the increase in data volume and available computing power, recent non-factoid Question-Answering work also achieved remarkable results using Deep Learning techniques. Hence, work done on non-factoid Question-Answering tasks using Deep Learning techniques, especially answer ranking tasks, is covered next.

Following the review of existing answer ranking systems and non-factoid QA, an overview of the current Question-Answering tasks related to sentiment analysis is conducted. Several state-of-the-art sentiment analysis frameworks are also reviewed and compared. Finally, the evaluation metrics used for this study are reviewed.

2.2 Existing Question-Answering datasets

The Multi-perspective Question-Answering (MPQA) Opinion Corpus (Wiebe et al., 2005) is one of the open public datasets for Question-Answering tasks, especially for Question-Answering related with user opinion mining, which is also related with the sentiment. The "MPQA Corpus contains 535 documents from the world" (Wiebe et al., 2005), containing information on various topics. All the documents in this dataset are "marked with expression-level opinion annotations"

(Wiebe et al., 2005) as tags. This dataset cover news articles, and other manually annotated text documents for users' opinions and other personal states, for example, sentiments, beliefs, emotions, speculations, and so on.

The Stack Exchange dataset ("Stack Exchange Data Dump: Stack Exchange, Inc.: Free Download, Borrow, and Streaming", 2019) is an anonymized data dump of information that users contributed to the Stack Exchange network. Stack Exchange network is one of the biggest Community Question-Answering (CQA) neighborhood. It includes 94 different websites for Question-Answering on different topics. Each website has its coverage of questions, answers and user comments in some specific domain. Each site includes information like "Posts", "Users", "Votes", "Comments", "PostHistory," and "PostLinks". This dataset is available on the website for free.

Yahoo! Answers Manner Questions dataset (version 2.0) is a dataset from Yahoo research. Yahoo! Answers is a website where people can post their questions online, and answer existing questions posted on this website. All the questions and answers are public to the user visiting this website. It is a small subset of questions, selected for their linguistic properties. For example, they all start with the word "how," following any word from the list: "to," "do," "did," "does," "can," "would," "could," and "should". The data contained within this dataset is annotated with labels such as "subject," "content," "bestanswer," "cat," "nbestanswers," "maincat," "subcat," "yid," "best_yid," "uri". There is one 'bestanswer' for each question.

2.3 Existing non-factoid Question-Answering systems

Most of the state-of-the-art Question-Answering (QA) systems are designed for answering fact-based questions such as "When was Steve Jobs born?" and "Who is the current president of the US?" In addition to facts, in various scenarios, people sometimes would "like to know about others' opinions, ideas, and feelings of some specific topics" (Oh et al., 2012). One category of Non-factoid questions: opinion questions, aim at revealing people's opinions. The answer to those questions could have long answers compared to fact-based factoid questions. Examples of non-factoid questions: "how can I get proper jaw cut, n loss weight?" "Traditional QA approaches are not sufficient enough to retrieve answers for opinion questions" (Stoyanov et al.,

2005 to match accuracy of factual questions, but non-factoid Question-Answering tasks have made some significant progress, which is discussed below.

2.3.1 Some early attempts for non-factoid QA

In the field of non-factoid Question-Answering, some research has been conducted to incorporate user opinion, attitude, or sentiment into the Question-Answering systems.

Early attempts in this domain include opinion analysis from pure text-based data to that in Question-Answering systems. Stoyanov et al. (2005) described OpQA, a corpus of opinion questions and answers, and compared different properties of fact and opinion questions and answers based on the OpQA corpus. According to the "disparate characteristics of opinion vs. fact answers" (Stoyanov et al., 2005), they conclude that traditional fact-based QA approaches could have difficulty in Multi-Perspective Question-Answering (MPQA) tasks without modification. Thus, Stoyanov et al. (2005) developed opinion summarization by employing machine learning approaches and rule-based subjectivity and opinion source filter on the Multi-Perspective Question-Answering system, which aims to identify the opinion-related information that the user exposed in a question.

Somasundaran et al. (2007) explored employing "attitude types for improving Question-Answering (QA) on both web-based discussions and news data" (Somasundaran et al., 2007). They researched "a set of attitude types developed with an eye toward QA" (Somasundaran et al., 2007). Using the attitude annotations, Somasundaran et al. (2007) developed automatic classifiers for recognizing sentiment and arguing attitudes. Finally, identifying the information of those attitude types of questions and answers showed positive results for improving opinion QA performance.

The methodologies described are still focused on rule-based or human-annotated approach, but this progress set a good foundation for non-factoid Question-Answering tasks, especially for non-factoid QA with sentiment/opinion into consideration. With the advancement of computing power and available data, it is becoming promising that more complex tasks of non-factoid QA with sentiment/opinion can be tackled with learning approaches.

2.3.2 Non-factoid Question-Answering with answer ranking and selection

Answer selection is an important procedure in QA system. It is also a crucial task with applications in information extraction and information retrieval. However, in real-world practices, the correct answer might not directly have similar lexical units overlap with the question. Instead, they could only be related semantic wise. For example, the topic addressed could be internally related, like cooking and recipes. The difference in lexical units is the unique feature of non-factoid Question-Answering compared to factoid Question-Answering. Some recent progress in this field is addressed below.

Up to this point, deep learning models have achieved "significant success on various natural language processing tasks" (Tang et al., 2015), for instance, "machine translation" (Bahdanau et al., 2015) as well as "text summarization" (Rush et al., 2015). Using deep learning approach to perform Question-Answering tasks is considered feasible.

Previous work on answer selection usually adopted approaches like "feature engineering, linguistic tools, or external resources" (Stoyanov et al., 2005). For example, lexical semantic resources were leveraged, and "semantic features were constructed based on WordNet" in (Yih et al., 2013). Based on word semantic relations, this model put related words in pairs based on semantic relationships. By following the word-alignment paradigm, Yih et al. (2013) found that the rich lexical semantic information could improve the model's performance consistently upon "the unstructured bag-of-words" (Yih et al., 2013) set-up, and also the ability for the model to learn latent structures. They concluded that adding shallow semantic information is more effective than introducing complexly structured constraints in answer selection tasks.

In other research, "the answer selection problem is transformed into a syntactical task" (Wang & Manning, 2010), performing "matching tasks between the question-answer pairs parse trees" (Wang et al., 2007). They presented a syntax-driven approach for Question-Answering tasks, explicitly aiming at solving the short-answer selection tasks for questions. Instead of directly using syntactic features to augment existing statistical classifiers (as those work mentioned above), they assume that questions and their (correct) answers have an inner relationship between each other via predictable syntactic transformations. Except for research based on

syntactical transformations, some work used “minimal edit sequences between those dependency parse trees for question-answer matching” (Yao et al., 2013). Recently, “discriminative tree-edit features extraction and engineering over parsing trees” was automated in the study presented by Severyn and Moschitti (2013).

Although approches employing syntactic features and dependency parse trees show good performance in non-factoid QA, they might suffer depending on available additional resources, or the results of feature engineering, or the complexity caused from employing linguistic tools.

Other research conducted in the domain of non-factoid Question-Answering systems is using traditional Natural Language Processing and text mining techniques. For example, sentiment analysis and spell checking, and also social network behaviors like votes with user information to predict the best answers (Eskandari et al., 2015). This work considers comments as one of the inputs, combining with other features mentioned before, by finding the combination of different features that works best for this model, their performance of the model shows improvement.

Other than research on feature engineering (Eskandari et al. 2015), previous approches were using deep learning techniques for the answer ranking task. The approaches for non-factoid QA usually belong to these directions: firstly, “the question and answer representations are learned and matched by specific similarity metrics” (Feng et al., 2015; Yu et al., 2014). Secondly, “a joint feature vector is constructed from both the question and the answer” (Wang & Nyberg, 2015). Then a classification task is performed on the joint vector, with the prediction being based on the ranking results from the joint feature vector (Wang & Nyberg, 2015). Similarly, “recently proposed models based on textual generation can be intrinsically used for answer selection and generation” (Bahdanau et al., 2015; Vinyals & Le, 2015). Using the given previous sentence or sentences in a conversation, they used a sequence-to-sequence framework to predict the next incoming sentence. It can be trained in an end-to-end fashion, thus this kind of model requires less hand-crafted rules. It is interesting to see that using this straight forward network structure, the model could find a solution to a technical problem via conversations on a specific domain, like IT helpdesk. On an open-domain movie transcript dataset, which is noisier, their model can perform common-sense reasoning in a simple form as a Question-Answering task. On the other

hand, it is also found that this model was lacking in consistency, which is a common failure mode for this research.

Semi-supervised approaches are also applied in non-factoid Question-Answering. Why-QA which is designed to retrieve answers from a given text passage. Examples of questions are "Why are tsunamis generated?" (Oh et al., 2012). Oh et al. (2012) adopted a machine-learning approach with a supervised classifier such as Support Vector Machine for answer ranking, and this research "successfully improved the why-QA performance" (Oh et al., 2012). To the best of our knowledge, this is the first work that considered sentiment analysis into the domain of non-factoid Question-Answering. "For the given pairs of a question and an answer candidate passage, the classifier arranges them into correct pairs and incorrect pairs, or gives a score indicating the likelihood that the pair is correct, which is used for ranking the answer candidates" (Oh et al., 2012). The hypothesis is that "given such a causal relation and a question generated from its effect part, the correct answer passage to the question has and indeed must have a substantial vocabulary overlap with the cause part of the causal relation" (Oh et al., 2012). This hypothesis is applicable to answer retrieval from a large-scale web corpus, but probably not for subjective questions, as these question-answer pairs could have less vocabulary overlap between question and candidate answer.

To the advantage of accessible computing power and increasing volume of data, the latest non-factoid Question-Answering work also achieved significant results using Deep Learning techniques. Tan et al. (2016) developed an approach with a neural network based on bidirectional LSTM. This deep learning model aims at non-factoid answer selection between question and answer pairs. Similar to the idea of Wang & Nyberg (2015), the prediction is based on the "joint feature vector based on both the question and the answer" (Tan et al., 2016). Instead of directly ranking similarity of the joint vector, as an improvement, the authors add CNN filters after the biLSTM hidden states to exploit more long-range sequential context information. Then the hidden states follows a max pooling layer for comparing similarity. The new network with CNN filters results in 3.7% higher accuracy over the selected baseline, based on InsuranceQA dataset (Feng et al.,2015), and 1.06% higher accuracy on TREC-QA dataset over various selected baseline models (Wang et al., 2007)

Tran et al. (2018) proposed another improvement over his biLSTM/CNN structure by adding attention system into the originally biLSTM framework presented in (Tan et al., 2015), aimed to learn low-dimensional vectors features. This network ″Obtain comparable performances to state-of-the art approaches" (Tran et al., 2018).

2.4 Deficiency in non-factoid Question-Answering using sentiment information

In the natural language processing field, much research has been done on sentiment classification, but so far there is research done in combining the sentiment information with non-factoid Question-Answering is quite limited. Sentiment information includes the user's sentiment polarity, opinion, subjectivity, and so on. An overview of existing research is summarized below.

Ku et al. (2007) presented an Opinion Question-Answering framework that aims at question analysis and retrieving answers from passage. They conclude that the best answers sometimes have sentiment correlation with the question (Ku et al., 2007). For opinion answer passage retrieval, they consider not only the relevance but also the sentiment contained. Considering opinion and action words together performs better than considering only opinion words.

Eskandari et al. (2015) proposed a design for predicting the best answers in Community Question-Answering systems based on sentiment. In this experiment, the Sentiment Analysis (SA) and subjectivity/objectivity identification are used to classify a given text into positive, negative or neutral and classes objective or subjective, which is of common practice for nowadays sentiment analysis framework. For SA, the text polarity gives a floating number within the range of [-1.0, 1.0], from the most negative, to the most positive. 0 stands for a neutral idea. For each entity (Answers or Questions), the sentiment analysis is performed upon its comments, which is their method to find users' opinion. They add average answer polarities, answer subjectivity, and Answer Comments Average Subjectivity together with other heuristic features such as Length of Answer, Answer count, resulting in 23 features together. Then decision tree classifiers are employed to perform classification.

In the domain of non-factoid Question-Answering related with sentiment analysis, some research has been conducted to extract the target aspects that the user addressed from the question, like attributes or components of the target product mentioned in the question, so as to extract the best answers in a more effective manner (Moghaddam et al., 2011). Another work was focused on using sentiment polarity of opinion to find the user intention in the question – what does the user focus on when asking this question (Mishra et al., 2016). Examples like: "I need a mobile with good camera and nice sound quality. WHY should I go for buying Nokia over Samsung?" The authors found the focus with the positive intention of buying Nokia rather than Samsung. This work focused on finding the emotionally supported objects, but did not address specific features (camera and sound quality), while the previous research does work on extract target feature component.

The first work introduces sentiment analysis to non-factoid Question-Answering by using sentiment analysis and word classes for ranking answers to WHY-questions in Japanese (Oh et al., 2012). This research generated sentiment information for word polarity and phrase polarity. Also, it "gains 15.2% improvement in precision at the top-1 answer" (Oh et al., 2012) over the baseline state-of-art QA system at that time. This research indicates that in the domain of open-ended questions, using sentiment and other Natural Language Processing features can achieve a likely gain in QA systems compared to simple fact-based Question-Answering without using the sentiment.

2.5 Existing sentiment analysis framework

This section introduces sentiment approaches. Kim and Hovy (2004) reported a system that determines word sentiment and combined sentiment of a sentence. Pang et al. (2002) classified documents by the overall sentiment rather than topic, with the polarity of a review being determined by the document's sentiment score. For opinion related text, Wiebe et al. (2002) invented a method for opinion summarization. In contrast to sentiment analysis based on word-level, Wilson et al. (2005) presented a phrase-level SA method which could identify the contextual polarity automatically. Ku et al. (2006) proposed a method to locate, collect, and organize different opinions from assorted information sources.

More recently, integrated frameworks have been proposed. The Stanford CoreNLP toolkit (Manning et al., 2014), is a framework that provides core natural language analysis. This toolkit is used in the academic field, with some research NLP community, and users of open source NLP technology. This framework included functionality of the parser for sentences, the part-of-speech tagger, the named entity recognizer, and sentiment analysis. For sentiment analysis, with the pre-trained model, it can categorize the sentence into categories like ‘positive’, ‘neutral’, and ‘negative’.

There are papers that conducted the state-of-the-art sentiment analysis framework comparison. In order to be useful, they excluded commercial use sentiment analysis frameworks, and domain-specific sentiment analysis frameworks. Zimbra et al. (2018) highlight the following frameworks:

- Sentiment140, with a classification accuracy: 66.46, providing binary (positive/negative) sentiment representations.
- SentiStrength, with a classification accuracy: 67.49, providing binary (positive/negative), trinary (positive/negative/neutral) and single scale (-4 to +4) representations.

These sentiment analysis frameworks provide cross-domain, sentiment classification ability, with sentiment measurement scale from 2-dimension (positive/negative), to 3-dimension (positive, negative, neutral).

Other choices include the highest ranked sentiment computation library for sentiment analysis framework on GitHub, Sentiment - AFINN-based sentiment analysis (Nielsen et al., 2011). AFINN is a kind of affective lexicon. This framework uses AFINN as a wordlist. Those words are rated for valence with an integer between minus five (negative) and plus five (positive). The library will return a comparative score, which is a sum of each token valence/number of tokens, positive words and negative words as a result.

Some sentiment analysis frameworks do not provide sentiment classification. Instead, they provide sentiment information in an extended scale (Felbo et al., 2017). They avoid the limit of the scarcity of manually annotated data. Compared to previous research used binarized emoticons like ‘positive’ and ‘negative’, and other specific hashtags in a form of distant

supervision, they extended it to a more diverse set of noisy labels of 64 dimensions. The models can learn richer representations compared to that sentiment information containing only positive/negative/neutral categories. They trained the model for emoji prediction based on 1246 million tweets containing one of 64 popular emojis. In the end, they obtained state-of-the-art performance in emotion and sarcasm detection using a single pre-trained model. This model is also widely applied to various research. Their results and analyses prove that "the diversity of emotional labels yield a performance improvement over previous distant supervision approaches" (Felbo et al., 2017) in the sentiment analysis field.

2.6 Evaluation Matrix

The following metrics can be used for evaluation: precision at 1; precision at k; mean average precision, and mean reciprocal rank.

Precision at 1 (P@1) is the mean of the precision of the top-ranked document retrieved calculated over all topics. When performing information retrieval tasks, one aspect of the evaluation is if the first relevant document is listed in the first place in the rank. It takes the first element in the result list and checks if this document is relevant. For this reason, P@1 has a value of either 0 (first document irrelevant) or 1 (first document relevant). Turpin et al. (2006) found that commonly reported measures, specifically mean average precision (MAP), do not usually provide good user performance on simple information retrieval tasks. They suggest that measures such as precision at 1 (Precision@1) could reflect actual user performance better (Turpin et al., 2006).

Precision at k (P@k), similar to Precision at 1, is the mean, which is calculated over all topics, of the precision of the first k documents retrieved from the task. This metric is appropriate for answer retrieval tasks that retrieve the answer from the entire paragraph or passage, and the correct answer count is more than one. Alternatively, in the context of recommendation systems, the users are most likely interested in the top-N items recommended by the system. In this case, it makes more sense to compute precision and recall metrics in the first N items instead of all the items.

Mean average precision (MAP) is the mean of average precision calculated over all topics. Although the MAP has been widely accepted as one of the standards for evaluation of information retrieval systems, it does not necessarily reflect precisely on how users perform on search tasks. This particular concern arrises because Thom et al. (2007)'s results show that the correlations "between the two categories of metrics are weak – therefore, the relative ordering of systems that are commonly used in the TREC framework may not reflect user performance" (Thom et al., 2007).

Mean reciprocal rank (MRR) is a statistical measure for evaluating any tasks which produce a list of possible responses according to a set of queries, which were ordered by probability of correctness. More specifically, the reciprocal rank of a query response is the multiplicative inverse of the rank of the ground-truth correct answer. For example, 1 for first place, 1⁄2 for second place, 1⁄3 for third place and so on. The mean reciprocal rank (MRR) is the numerical average of the reciprocal ranks of all the results for a set of queries. MRR is useful for measuring the ability of a system that performs well at finding one relevant document highly ranked in the ranking list (Thom et al., 2007).

2.7 Summary

The section described relevant research which is related to non-factoid Question-Answering, and sentiment analysis. It goes through the datasets specifically for non-factoid Question-Answering, existing Non-factoid Question-Answering framework based on traditional lexical approach and deep learning approach. Then moving on to current state-of-the-art sentiment analysis framework and finishing with the evaluation metrics used.

However, none of the work done concentrates on applying sentiment information to a non-factoid Question-Answering framework in English aiming at "How" questions, but rather using the lexical approach to perform the QA task. This is what this thesis aims to do – adding sentiment into non-factoid Question-Answering in English and test on "How" questions using deep learning.

CHAPTER 3. METHODOLOGY

This section describes methodology used in this research, namely, a dataset and data processing methodology; a framework for non-factoid Question-Answering system with sentiment and a baseline framework; and methods for evaluation.

3.1 Dataset

Our dataset is "L5 - Yahoo! Answers Manner Questions, version 2.0" from Yahoo research. Yahoo! Answers is a website where people can post questions and answers. The questions listed on the website are public to any web user accessing this website. The data contained in this dataset is extracted from the Yahoo! Answers corpus from a 10/25/2007 dump, it is a small subset of it. The questions contained are selected for their linguistic properties. For example, the questions always start with the word "how" following any word from the list "to," "do," "did," "does," "can," "would," "could," and "should". Questions and answers with apparently low quality are already dropped from the dataset; The kept ones have at least four words for this corpus. In those sentences, there is at least one noun and at least one verb.

There are 142,627 questions in this dataset, together with their answers, best answer selection, also category and sub-category of questions. The total size of this dataset is 104 Megabytes.

```
<subject>How do I deal with annoying little brother?</subject>
<cat>Men's Health</cat>
<maincat>Health</maincat>
<subcat>Men's Health</subcat>
<bestanswer>Why not suggest anger management for HIM? He probably has
resentment because you're older and you're smarter. You said you're
very smart, I'll bet dollars to donughts he resents that! My brother's
getting A's and B's and I'm getting C's and B's. Nevermind that my
brother is failing, he is still my parent's little angel. I hate my
brother, and I can sure sympathize with you! Anyway, the best thing to
do is just not listen to him. Talk to him when he wants attention. You
never know when someone's trying to get attention. That's all a lot of
bullys want, is love and attention.&#xa;&#xa;Good luck.</bestanswer>
<answer_item>Do this:&#xa;Dont tlk to him even if you need something-
trust me &#xa;If he needs a favor dont do it&#xa;most importantly
ignore him</answer_item>
```

Figure 1: Example of annotation

Due to computational complexity and speed of biLSTM model, only the first 30% of the data is used for this study. The data is split into training, validation, and testing set. The number of questions in training set is 20965, for validations set is 4492, for the testing set is 4493. Detailed answer count distribution (number of answers per question) for the selected dataset is shown in Figure 2.

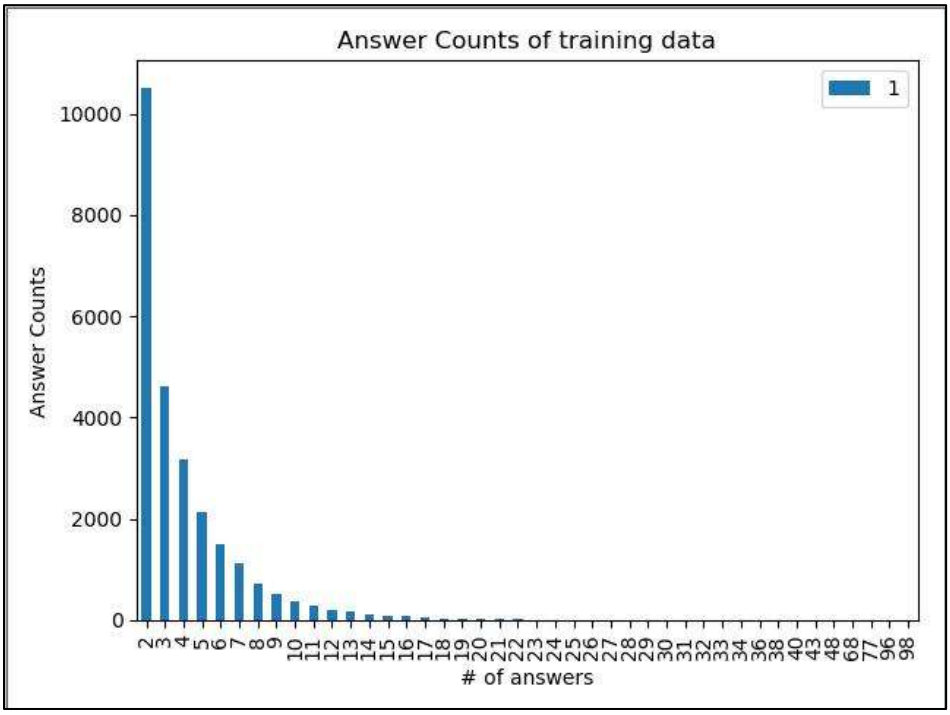

Figure 2 Distribution of answer count in training set

3.2 Research Framework

For this study, the answer ranking system (Tan et al., 2015) was adopted as a baseline. A new framework proposed in this work is based on this baseline and adds the sentiment as input to the model. Comparison between those two models was performed to validate whether adding sentiment can improve the performance of this network, as in the aspect of its ability to select the best answer from all the candidate answers.

The workflow for this method, shown in Figure 3, is described as follows:
Training: when given the question, the Question-Answering Framework should automatically select the best answers from the candidate answers provided from the dataset, then validate

whether it is selected as the best answer according to the annotation provided in the dataset. The network can automatically learn the pattern within the data, thus having a better accuracy predicting the best answer.

Evaluation: the best answer prediction process is conducted on both the baseline model and the proposed new network architecture. The performance between the two models was compared to determine adding sentiment information of questions and answers into prediction can improve the prediction quality of the best answer.

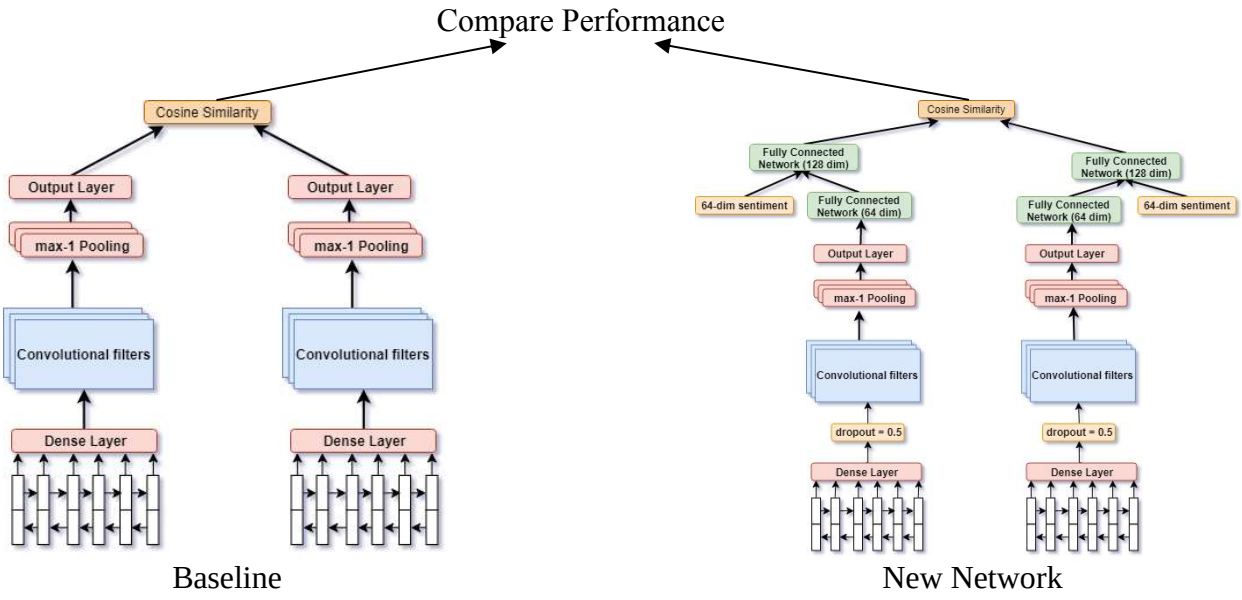

Figure 3 Workflow

3.2.1 Baseline Model

The baseline model (Tan et al., 2016) is a bidirectional LSTM network with a CNN on top of it. The network performed cosine similarity comparison to determine the similarity ratio between question and answer pairs. The answer with the highest similarity is selected as the best answer by model. The structure of this baseline model is visualized in Figure 4.

The LSTM is applied separately over Word Embeddings of question and answers to get a more precise representation of time sequences of sentences, creating hidden vectors for both question and answer. The hidden states of the network are inputted into a CNN architecture as the CNN architecture provides better performance in answer ranking tasks, especially for data with long answers. At the same time, it also "provides a more composite representation of questions and answers" (Tan et al., 2016). After applying four CNN filters with length of 1,2,3,5, max-1 pooling was applied to extract the features that are most significant. Compared with "evenly considering the lexical information of each token" (Tan et al., 2016), this architecture emphasizes certain parts of the answer, so as to be able to differentiate the incorrect answers with the ground truth answers.

Details about this CNN network is described as follows: "for every window with the size of m in biLSTM output vectors, Hm(t) = [h(t), h(t + 1); … ; h(t + m - 1)], where t is a certain time step, the convolutional filter F = [F(0) … F(m - 1)] would generate one value as follows" (Tan et al., 2016).

$$\mathrm{OF}(t) = \tanh[(\sum_{i=0}^{m-1} h(t+i)^T F(i)) + b] \quad (1)$$

In this function, "b is a bias, and F and b are the parameters of this single filter" (Tan et al., 2016).

Just as a typical CNN, a max-k pooling layer is applied on top of the convolutional layer. After maxPooling, for each filter applied, the maximum values are kept, which allows the information that matches the input sequence to be kept with minimum information loss. In this study, k is selected as 1, since k > 1 did not lead to any significant improvement in early attempt of this experiment.

After generating the output of the CNN, a pairwise ranking method was adopted to define the objective function. For question and each of its candidate answers, a question-answer pair is constructed. The similarity was computed for each input pair. The answer with the highest similarity score is selected as the best answer.

During the training process, the loss is computed for adjusting the weight of this model. The loss is computed by the distance for each question-answer pair is computed as follows:

$$L = max\{0, \lambda - sim(q, a_+) + sim(q, a_-)\} \quad (2)$$

where a_+ is the ground truth, which can be thought of the answer with the annotation of "best answer" from the dataset, a_- is an incorrect answer from other candidate answers belonging to the same question. Tan et al. (2015) randomly selected 50 answers from the entire answer space as incorrect answers, which means the answer space collected all answers from different questions in the entire dataset. For our study, the incorrect answer is selected from the answer space under the same question as an improvement. This practice could make the model learn the relationship specifically between the question-answer pairs belonging to the same topic. The baseline model only uses the answer pair with the highest L for updating the weight.

The network structure of the baseline is as follows:

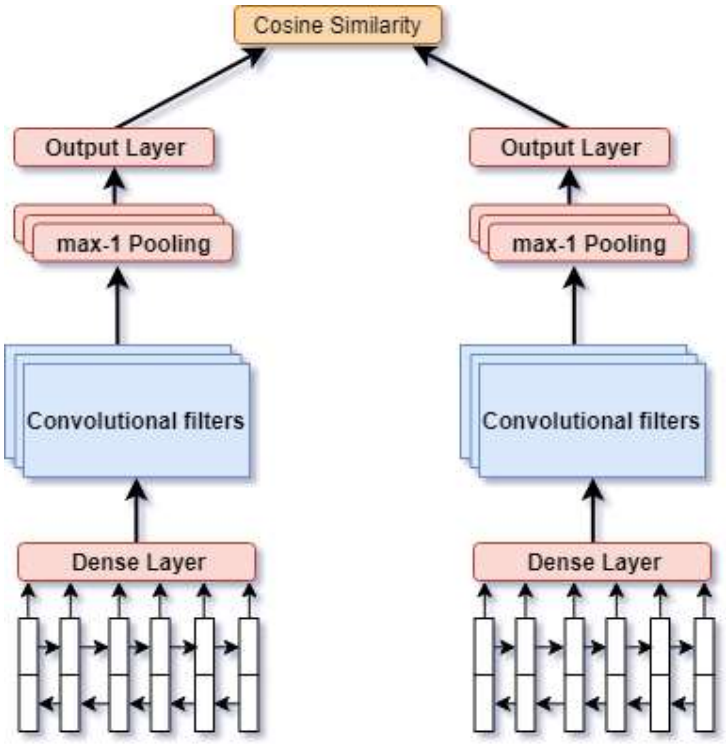

Figure 4 Baseline Model Structure

3.2.2 Sentiment Information Computation

For this study, the sentiment information is computed using the pre-trained model from Deepmoji (Felbo et al., 2017). Compared to dividing sentiment information into three scales (positive, negative, and neutral), Deepmoji extended "the distant supervision to a more diverse set of noisy labels" (Felbo et al., 2017), more specifically, 64 kinds of emojis. In this approach, the models can learn richer representations. The pre-trained model is trained on 1246 million tweets containing one of 64 common emojis. This model obtained "state-of-the-art performance" within sentiment detection (Felbo et al., 2017).

The input to this pre-trained model is the questions and their corresponding candidate answers. The output for each input sentence is a 64-dimension vector, representing the confidence of each emoji. An example of the corresponding input and output of this model is provided in Figure 3, here we only visualize the top-5 emojis with the highest confidence scores.

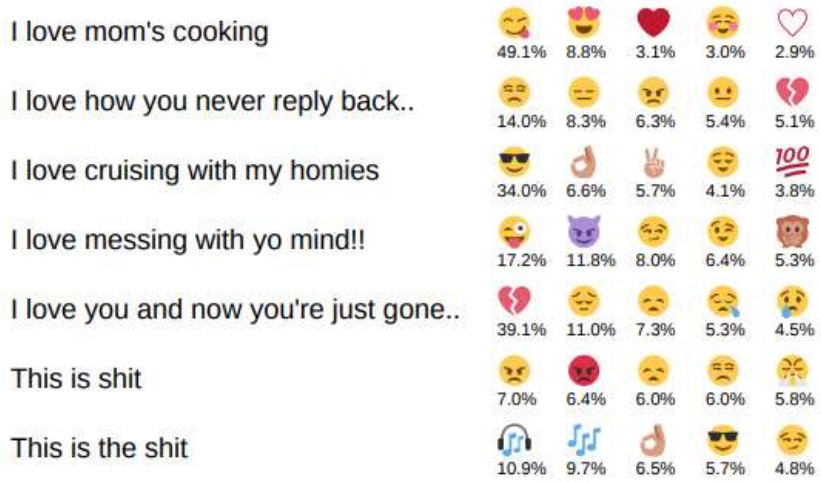

Figure 5 partial example outputs of Deepmoji (Felbo et al., 2017)

The 64-dimension sentiment information is generated for each question and answers in this dataset, stored to be used as the input to the new network.

3.2.3 New network with sentiment information as an input

For this study, a new network is designed upon the existing baseline network to incorporate sentiment information when performing the best-answer selection. The baseline model aims at extracting information based on text, and it is desirable to keep this architecture in the new model.

Sentiment information is contained at a sentence-level, thus it is not added to the hidden states of biLSTM, which corresponding to word-level information. Instead in this model it is added as extra information after the text information is computed and adjusted. The only adjustment to this biLSTM-CNN model is adding a dropout after the biLSTM hidden states due to the high amount of data that was inputted into this model, which could cause overfitting. The dropout rate is selected as 0.5, as it effectively enables the model to sample from a probability distribution of the designed network architectures, and also it is generally used as an optimal amount for dropout in deep learning tasks.

In order to assign equal weights to text information contained in the question and answers, and the corresponding sentiment information contained within them, the output of CNN is passed into a Fully Connected Network, which is constructed of a Dense Layer, one activation layer with ReLU activation function, and another Dense Layer. The ReLU activation function is selected because ReLU offers faster converging speed, and also is more capable of reducing the possibility of vanishing gradients in the training process. The output of this Fully Connected Network (FCN) is a 64-dimension vector, which is the same dimension as the sentiment vector.

Then the 64-dimension vector of FCN is concatenated with the 64-dimension sentiment vector, resulting in a 128-dimension joint vector. This joint vector is passed to another Fully Connected Network (FCN), with the output being a vector of 128-dimension vector. The size 128-dimension is chosen with the concern of reducing dimension sometimes leads to information loss. Thus, the dimension reduction should not be over-done. This Fully Connected Layer (FCN) can make the model learns the text feature and sentiment feature together, in the process of tuning the network weight.

After generating the 128-dimension vector representation, similar to the baseline, a pairwise ranking method was adopted. For question and each of its candidate answers, a question-answer pair is constructed. The similarity was computed for each input pair, and the difference is the similarity is based on the second FCN output, not the direct output of the max-1 pooling layer as in the baseline. The answers with the highest similarity score are selected as the best answer. The model architecture is shown in Figure 6.

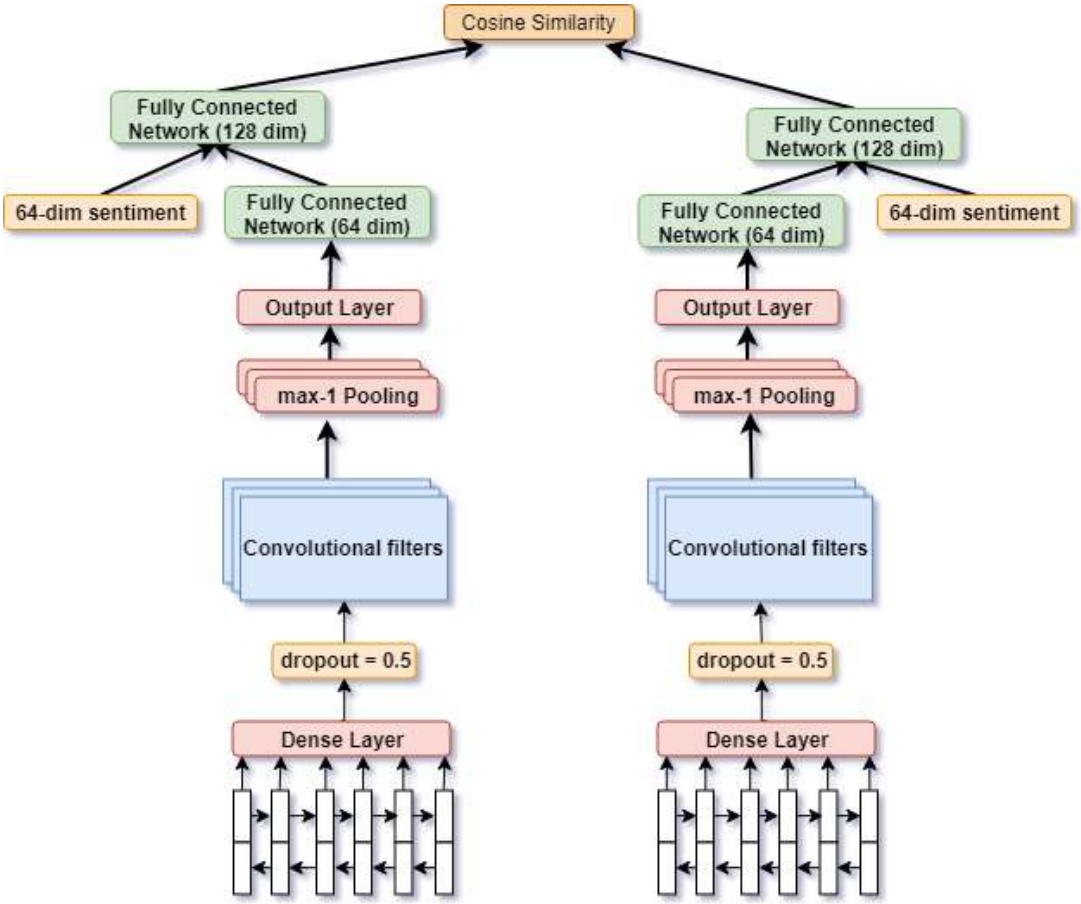

Figure 6 The proposed model structure

3.3 Evaluation

Evaluation was performed by P@1 (Precision of the top answer), and MRR (Mean reciprocal rank). Precision is not chosen because there is only one correct ground-truth answer. MAP is not chosen because MAP is usually used when performing information retrieval on multiple topics, and an average of the precision is required across different topics. For this task, all the candidate answers under the same question belong to the same topic, so MAP is not appropriate for this study.

P@1 is the precision of the top answer, measuring how many questions have a correct top answer candidate. In this case, since we only have one predicted best answer, and only one ground-truth best answer, P@1 is also equal to accuracy in this case.

MRR stands for Mean reciprocal rank. This matrix is used for "evaluating any process that produces a list of possible responses to a sample of queries, ordered by probability of correctness" (Thom et al., 2007).

$$\text{MRR} = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \tag{3}$$

3.4 Experiment Details

For Natural Language Processing tasks, a typical experiment streamline could be a process like the following graph.

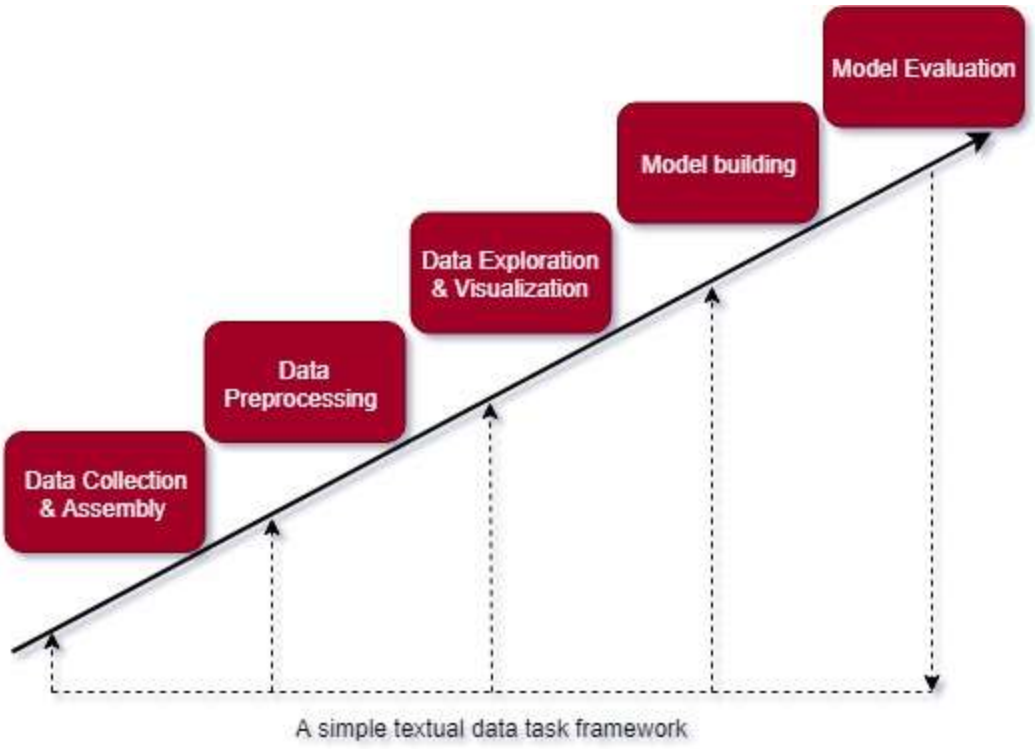

Figure 7 Workflow for a typical NLP task (Mayo, M., 2017)

The implementation details of this experiment are described in this section. We already described the chosen dataset in section 3.1, with data visualization of its answer count distribution, and average answer count. In the following sections under 3.4, experiment details about Data processing, experiment set up, parameter setting, network structure, and the training and testing process will be described.

3.4.1 Data Preprocessing

The data from the Yahoo! Answers Manner Questions dataset is in a format of XML file. It contains information tagged such as “subject,” “content,” “bestanswer,” “cat,” “nbestanswers.” Data belonging to those specific categories, which is needed for the model should be extracted and cleaned to meet the requirements.

3.4.1.1 Extract data from the original dataset

Among existing tags such as “subject,” “content,” “bestanswer,” “cat,” “nbestanswers,” “maincat,” “subcat,” “yid,” “best_yid,” “uri.” The information needed for this study lies in the

category of ″subject”, which means questions; “bestanswer”, corresponding to the answer marked as the best answer among all the candidate answers; the “nbestanswers” tag has several sub-tags under it, the sub-tag is called “answer_item”, which corresponds to one candidate answer item. This information is needed for this research.

After extracting these tags, for each question, the data is transformed into the following representation:
“Question,” “Best answer,” “list of candidate answers.”

3.4.1.2 Data cleaning

For Natural Language Processing tasks, data cleaning is a crucial process. Bad data could lead to incorrect tokenization, which results in an imprecise experiment result. For the information extracted in the previous step, the following data cleaning process were performed.

In some data cleaning tasks, punctuations are removed completely, since those studies aims at the lexical information contained in sentences. In this study, the punctuations were kept due to the importance of punctuations in sentiment analysis. For Deepmoji (Felbo et al., 2017) sentiment framework, emoticon like: ′;-)’, ‘;o)’, ‘(:’, ‘:o)’ would make a difference in the final sentiment distribution of the sentence. Repeated punctuations like ‘!!!!!!!’ would also significantly affect the results.

Non-ASCII characters were removed, enabling to focus on text that contains valid information. Additionally, phone numbers, continuous white spaces, and URLs were also removed. Some words are connected by ’.’ as a mistake, thus leading to incorrect tokenization. For example, the tokenized result will contain records like ‘kitchen.I’. This scenario is also corrected by separating the previous word and the latter word with ‘. ’.

There are other scenarios when a word is connected with hyphens on one end or both ends, leading to tokenization results like: ‘nonrainy-,’ ‘-obey.’ In this case, the hyphens were removed. Continuous hyphens with different length were also removed due to replication in tokenization.

3.4.1.3 Split dataset

To the concern of the size of this dataset, and the computation expensiveness of LSTM networks, 30% of the entire dataset is adopted as the data used in this study. The <Question> - <Best answer> - <list of candidate answers> records are divided into train, validate, and testing set by a ratio of 0.7, 0.15, and 0.15, as a classic partitioning ratio in deep learning field.

3.4.2 Text encoding

For Natural Language Processing tasks, text from the data should be tokenized and encoded before feeding to the model. For this study, the data preprocessing process is similar to the baseline experiment (Tan et al., 2016). The dictionary of the entire data space was generated by Natural Language Toolkit (NLTK). Tokenization was performed on the training, validation, and testing set, using TweetTokenizer from NLTK. The final encoded data was generated based on the dictionary id. Questions and answers are padded into a length of 150, and if the length of the question/answer is longer than 150, only the first 150 words are selected.

3.4.3 Sentiment Information Computation

"The state-of-the-art deep learning framework" (Felbo et al., 2017) for analyzing sentiment was used for this study to achieve the sentiment information corresponding to each question, and answer in the dataset.

For each paragraph, according to question or answer for this dataset, the framework will generate a 64-dimension confidence list corresponding to 64 different emojis. The full list of emojis can be found in the Deepmoji (Felbo et al., 2017). As the experiment iterated over the dataset, corresponding sentiment information aligning with the text of each question/answer were received. The result for sentiment information was inserted into the dataset. The final data format is as follows:

- "Question," "Question_sentiment," "Best answer," "list of candidate answers," "Answer_sentiment".

3.4.4 Baseline model training process

The baseline model is trained on 20965 questions and their corresponding answers. The models in this study are based on Tan et al. (2016). We use the accuracy matrix, in this case, also Precision at the top one answer, to check the performance of the model on the validation set to locate the hyper-parameter setting for this experiment, also the best epoch.

The model is trained in a batch of 64, and "the maximum length L of questions and answers is 150. Any tokens out of this range was discarded" (Tan et al., 2016). The initial word embedding was trained by employing word2vec (Mikolov et al., 2013). The word vector size was set to 100. Word embeddings are also parameters and were optimized according to input data during the training process. Rmsprop is the optimization strategy. The margin values are set to 0.2.

Question and answer input, which were already encoded, were fed into the encoding layer. Then the embedding layer were fed into biLSTM, the hidden state of biLSTM was passed through CNN and max-1 pooling layer.

The LSTM hidden state vector was selected to be 200-dimension for one direction, for biLSTM, after concatenating the output vectors, the hidden state output shape was a 400-dimensional vector for each word. The CNN windows size was set to 1, 2, 3, 5, the output of each CNN window after max-pooling was a 500-dimension vector, which was concatenated in the end, resulting in a 2000-dimensional output to be used for comparing the similarity between questions and answers.

3.4.5 New network structure construction

For this study, the way how sentiment information is inserted into the network is crucial. Sentiment information was generated based on the entire paragraph, which corresponded to one question/answer, hence the sentiment information was a sentence-level/paragraph-level information compared to word-level information. In this case, inserting sentiment information into each word vector did not make sense, thus we kept the original LSTM/CNN structure for generating a vector representation of question/answer, and added sentiment information after it.

The only modification was to add a dropout layer after the LSTM hidden state layer preventing the overfitting problem, already described in the previous section.

In order to give the sentiment information more weight, we passed the output of max-pooling layer to a Fully Connected Network (FCN), reducing it to a 64-dimension vector, which was of the same dimension of sentiment vector, thus giving those two vectors the same weight during the training process. This 128-dimension joint vector is passed into another FCN to learn sentiment information pattern contained within the data, leading to an output of shape 128 dimensions, which was used for similarity comparison between question and answer in the last step.

The first Fully Connected Network (FCN) was initialized with an input shape of 2000, for each input element. The output shape was set to 64 for each element, and the total element counts depended on the number of question/ground-truth-answer/wrong answer pair passed to the network. The activation function was set to ReLU, the bias is initialized to constant 0.1 as when using ReLU, it is a usually good to initialize them with a positive bias to reduce the chance of having "dead neurons". The second Fully Connected Network (FCN) was of the same setting except that the input shape for each element was 128, the output shape was also 128 per element.

3.4.6 New model training with data combined with sentiment information

The sentiment information is passed through the 64-dimension sentiment input layer. The total question passed for training was 20965, the best answer was 20965, wrong answer count is 106733. Those question/answer used for training the new network were the same as those used for training the baseline. The average answer count is 4.13 excluding the question with only one answer.

The accuracy matrix was used to check the performance of the new model on the validation set to locate the best hyper-parameter settings and the best epoch. The final performance was tested on the testing set. The new model was trained in a batch size of 64, and the maximum length L of questions and answers is 150. Any tokens out of this range was discarded. The dropout rate for the dropout layer after the LSTM hidden state layer was set to 0.5. The rest of the experiment settings was the same as the baseline model.

3.4.7 Performance evaluation between baseline and new model

For this study, the evaluation matrix used is Precision@1 and MRR. Since there is only one predicted best answer, and only one ground-truth best answer, P@1 is also equal to accuracy in this case.

The new neural network with the sentiment information is evaluated on the validation set with 4492 questions and test dataset with 4493 questions, the Precision@1 and the Mean reciprocal rank (MRR) scores are calculated, and compared with the baseline performance. A paired T-Test are employed to calculate the statistical significance between the new neural network, and the baseline.

3.4.8 Testing of four categories

A further testing scheme was designed to test the performance difference between the neural network with sentiment and the baseline. This set of tests are focused on two variables: sentiment, and subjectivity. The hypothesis is that since subjective questions request a person's personal opinion, sentiment could play a heavier role for these questions. The new test suite is divided into four tests listed below:

- Questions without sentiment versus questions with sentiment.
- Subjective questions versus non-subjective questions.
- Subjective questions with sentiment versus all subjective questions.
- Non-subjective questions with sentiment versus all non-subjective questions.

In order to perform this set of tests, the dataset should be further annotated to select the subset for question-answer pairs whose question contains sentiment or does not contains sentiment, and also the subset of question-answer pairs whose question belongs to subjective questions or non-subjective questions.

For questions containing or not containing sentiment, the Stanford core NLP sentiment analysis framework (Manning et al., 2014) is employed. This framework allows classification of the input sentence into three categories: neutral, positive, and negative. In this experiment, the positive and

negative questions are regarded as ‘questions with sentiment,’ the neutral question is regarded as ‘questions without sentiment.’

For subjective question and non-subjective question, the Textblob (Loria et al., 2014) text analysis framework is employed, as it is used by several NLP papers (recent papers include Hasan et al., 2018), specifically to generate subjective scores of an input sentence as one of the input features of the Eskandari et al. (2015). With a subjectivity scale from 0 to 1, a question with subjectivity score bigger than 0.6 are considered a subjective question, at the same time, a question with subjectivity score less than 0.2 is considered a non-subjective question in this study.

The data used for this set of tests are acquired as follows. The entire test dataset is of size 4493. The ratio of the questions with sentiment is relatively small compared with the questions without sentiment (also regarded as neutral questions). The number of questions with sentiment is not enough for testing if the data is only retrieved from the testing set. The same circumstance applies to subjective questions.

To address this concern, the data used for this set of tasks are retrieved from the rest 70% of the dataset. The first 4000 questions for each category in sub-test one and two are selected and shuffled, ready to be used for the final testing. The questions in subjective classification with sentiment category are retrieved by performing the intersection of 4000 subjective questions and 4000 questions with sentiment. The questions in non-subjective set with sentiment are retrieved by performing the intersection of 4000 non-subjective questions and 4000 questions with sentiment.

The evaluation follows the same metrics as for the overall architecture performance (Precision@1 and MRR).

3.5 Summary

This section describes the methodology and experiment details used for this study. This includes the datasets being used, the preprocess conducted, and the detailed experiment settings. The test metrics employed are also described, after which the experiment details are elaborated upon.

CHAPTER 4. RESULTS AND DISCUSSIONS

Two evaluation metrics – Precision@1 and Mean reciprocal rank (MRR) are calculated for the entire testing set, as well as for the four tests with subjectivity and sentiment. The results are compared with the corresponding baselines.

For this study, each question has one labeled 'best answer,' and several candidate answers. The 'best answer' and other candidate answers composed the entire answer pool for each specific question. When predicting, the model would predict the best answer based on the confidence score generated, the answer with the highest confidence score is selected from the answer pool of answers for each question, and marked as the 'best answer.' The evaluation matrix is based on Precision@1, and MRR (Mean reciprocal rank). The improvement matrix shows the performance difference between the new network and the old one, and the calculation method is as follows:

$$\text{improvement} = (\text{new\_model\_score} - \text{baseline\_score}) * 100\% \quad (4)$$

4.1 Model performance comparison

In this section, the performance is compared between the proposed neural network architecture and the baseline. Each model is run separately on the validation set and testing set to get the results.

4.1.1 Performance comparison on the validation set

After splitting the dataset into three subsets, the training, validation, and testing set, the validation set resulted in 4492 questions with their corresponding answers and their sentiment information.

For validation set, after running the baseline of the biLSTM/CNN (Tan et al., 2016) network, and then the new neural network model with sentiment information, both evaluation metrics showed improvement over the baseline.

The results for the proposed new neural network architecture and the baseline are as follows (statistically significant results are marked with '*'):

Table 1: Metrics for the validation set

Metric	Precision@1	MRR
Baseline scores	0.4484	0.6387
New neural network scores	**0.5859***	**0.7504***
Improvement	13.76%	11.17%

For the new neural network, Paired T-Test is run to test for the statistical significance on the validation set. The validation set is split into five equal parts, on which both the baseline and the new neural network architecture is run. The results are as recorded below:

Table 2 Precision@1 metric for the validation set

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.6025	0.4025	0.4101	0.4097	0.4189
New neural network	0.8315	0.5122	0.5275	0.5220	0.5401

For the Precision@1 metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.0019, and the improvements seen are statistically significant.

Table 3 MRR metric for the validation set

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.7512	0.6019	0.6119	0.6080	0.6218
New neural network	0.9025	0.7117	0.7213	0.7227	0.7301

For the MRR metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.00007, and the improvements seen are statistically significant.

In conclusion, the statistically significant improvements are captured by Precision@1, MRR.

4.1.2 Performance comparison on the testing set

After splitting the dataset into three subsets, the training, validation, and testing set, the testing set was left with 4493 questions with their corresponding answers and sentiment information.

For the testing set, after running the baseline of the biLSTM/CNN (Tan et al., 2016) network, and then the new neural network model with sentiment information, the Precision@1 saw improvement, the MRR matrix performance does not see a significant difference.

The results for the proposed new neural network architecture and the baseline are as follows (statistically significant results are marked with '*'):

Table 4 Metrics for testing set

Metric	Precision@1	MRR
Baseline scores	0.5593	0.7395
New neural network scores	**0.5718***	0.7379
Improvement	1.25%	-0.16%

For the new neural network, Paired T-Test is run to test for the statistical significance on the testing set. The testing set is split into five equal parts, on which both the baseline and the new neural network architecture is run. The results are as recorded below:

Table 5 Precision@1 metric for the testing set

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.5194	0.5269	0.5372	0.6897	0.5220
New neural network	0.5371	0.5396	0.5491	0.6991	0.5290

For the Precision@1 metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.00139, and the improvements seen are statistically significant.

Table 6 MRR metric for the testing set

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.7151	0.7295	0.7270	0.8076	0.7188
New neural network	0.7138	0.7249	0.7269	0.8169	0.7064

For the MRR metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.31776, and the improvements are not statistically significant. However, Precision@1 showed statistically significant improvement.

4.1.3 Discussion of results of two category

For this test, the Precision@1 performance gain reached 13.76% for the validation set, and 1.25% for the testing set. While the baseline model performance varies by 11.10% between validation set and testing set, the new model's performance is both higher and more stable, which indicating the new model has better ability to generalize to more data, and have a more consistent performance.

While the performance gain between validation set and testing set is relatively big, further investigation into the question categories is conducted. The details of questions of each category are summarized and visualized in section 4.2.2. It is noticeable that the number of questions belonging to the non-subjective questions with sentiment, and subjective questions without sentiment category are higher than that of the testing set. Those two categories offer better performance gain overall (the reason for this conclusion can be identified from sub-test 3 in section 4.5.3). At the same time, the new model performed on subjective questions with sentiment does not improve or even decrease the performance compared with baseline (the reason for this conclusion can be identified from section 4.6.3), validation set has less subjective questions with sentiment than that of testing set, this could also be another reason why performance gain on validation set is much higher than that of testing set.

4.2 Data distribution of different question category in Sub-Tests

Four categories of test are performed:

- Task 1: questions without sentiment versus questions with sentiment.
- Task 2: subjective questions versus non-subjective questions.
- Task 3: subjective questions with sentiment versus subjective questions without sentiment.
- Task 4: non-subjective questions with sentiment versus non-subjectives questions without sentiment.

For tasks 1 and 2, there are 4000 questions for each category retrieved from the remaining 70% of the entire dataset, which was not used for either training, validating, or testing the model. In particular, for task 1, 4000 questions without sentiment and another 4000 questions with sentiment were selected; for task 2, 4000 subjective questions and 4000 non-subjective questions were selected.

For tasks 3 and 4, the subjective questions with sentiment category has 511 questions, retrieved by performing the intersection of 4000 subjective questions and 4000 questions with sentiment. The non-subjective questions with sentiment task has 436 questions, retrieved by performing the intersection of 4000 non-subjective questions and 4000 questions with sentiment.

4.2.1 Data distribution of different question category in the training set

The network was not retrained to perform the experiments. However, to understand what it could learn, the number of questions for different categories is provided

- the number of questions without sentiment is 18336;
- the number of questions with sentiment is 2626;
- the number of non-subjective questions is 13707;
- the number of subjective questions is 2429.

The data distribution for each category in the training set is described in Figure 8.

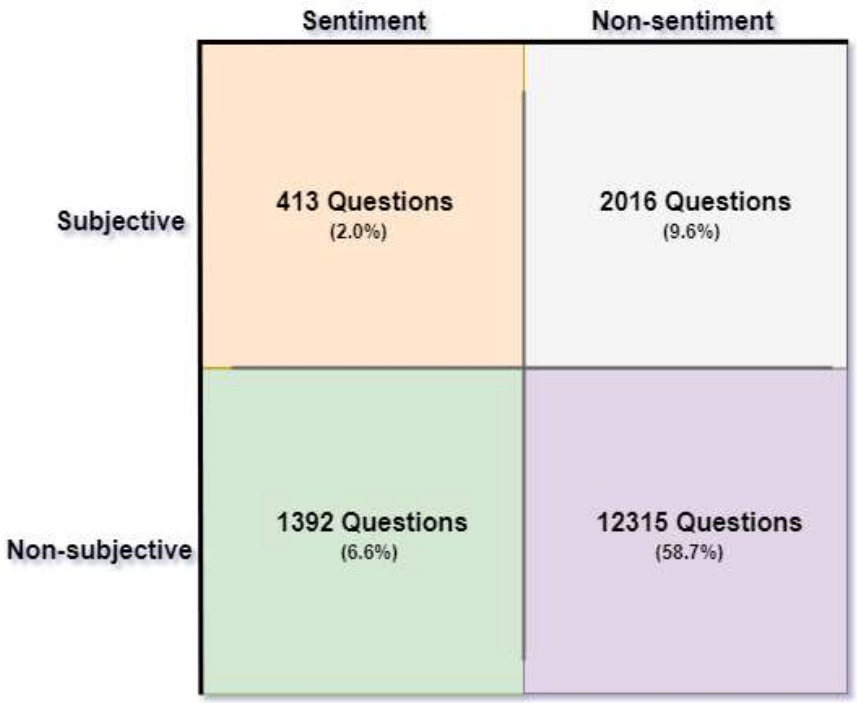

Figure 8 Data distribution for the training set

4.2.2 Data distribution for different question category in validation and testing set

The number of questions for different categories in the validation set is listed as follows:

- the number of questions without sentiment is 3938;
- the number of questions with sentiment is 552;
- the number of non-subjective questions is 2970;
- the number of subjective questions is 526.

The percentage of questions of a specific category as shown in Figure 9 is calculated as the number of questions in this category divided by the number of questions in the entire validation set.

The number of questions for different categories in the testing set is listed as follows:

- the number of questions without sentiment is 3912;
- the number of questions with sentiment is 582;
- the number of non-subjective questions is 3006,
- the number of subjective questions is 478.

The percentage of questions of a specific category as shown in figure 10 is calculated as the number of questions in this category divided by the number of questions in the entire test set.

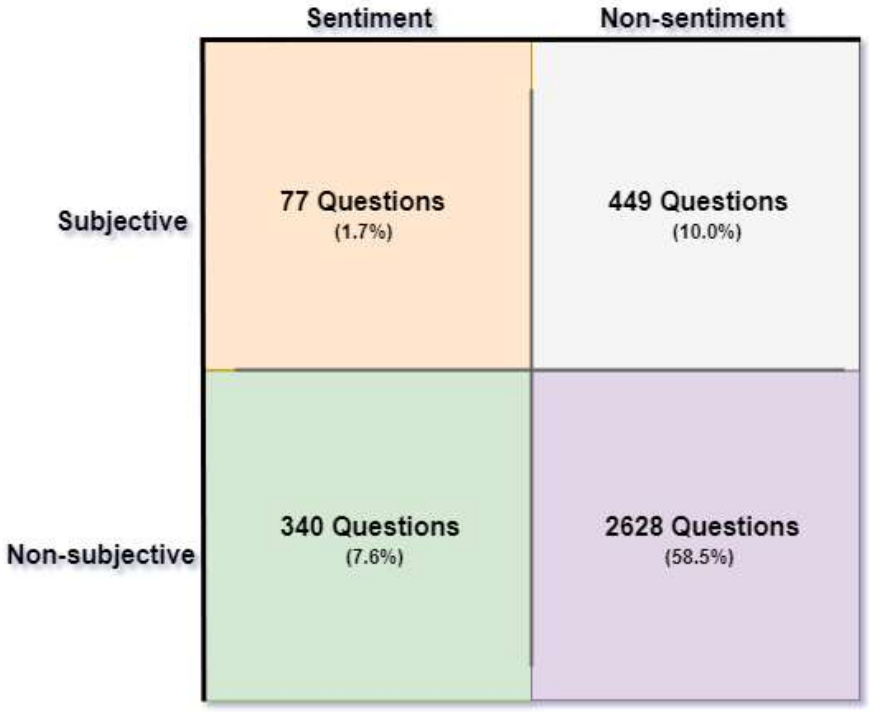

Figure 9 Data distribution for validation set

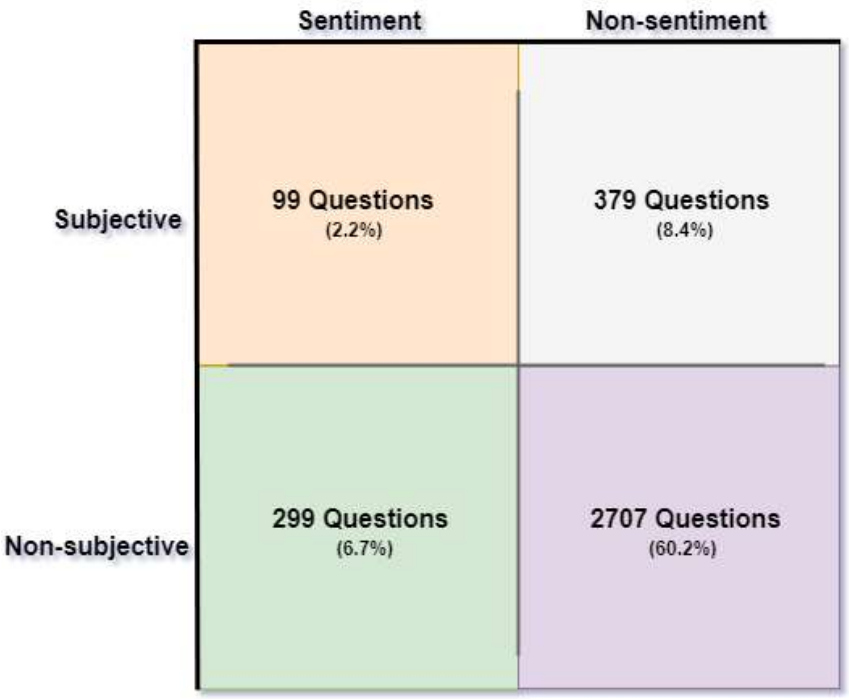

Figure 10 Data distribution for testing set

4.3 Sub-Test 1: questions without sentiment versus questions with sentiment

When pruning the dataset for questions without sentiment and questions with sentiment from the latter 70% of the entire dataset, the data for this sub-test is composed of 4000 questions without sentiment, and 4000 questions with sentiment, and their corresponding answers and sentiment information.

4.3.1 Results for questions with sentiment

After running the baseline of the biLSTM/CNN (Tan et al., 2016) network, and then the new neural network model with sentiment, both metrics showed improvement on this test, with data composed of questions with sentiment.

The results for the proposed new neural network architecture and the baseline are as follows (statistically significant results are marked with '*'):

Table 7 Metrics for test on questions with sentiment

Metric	Precision@1	MRR
Baseline scores	0.4616	0.6538
New neural network scores	**0.4758**	**0.6643**
Improvement	1.42%	1.05%

For the new neural network, Paired T-Test is run to test for the statistical significance on those questions with sentiment. The question-with-sentiment dataset is split into five equal parts, on which both the baseline and the new neural network architecture is run. The results are as recorded below:

Table 8 Precision@1 metric for questions with sentiment

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.4799	0.4688	0.4733	0.4421	0.4451
New neural network	0.5275	0.4659	0.4852	0.4510	0.4510

For the Precision@1 metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be .08801, and the improvements seen are not statistically significant.

Table 9 MRR metric for questions with sentiment

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.6777	0.6566	0.6583	0.6356	0.6423
New neural network	0.7091	0.6520	0.6734	0.6433	0.6454

For the MRR metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.07976, and the improvements seen are not statistically significant.

4.3.2 Results for questions without sentiment

After running the baseline of the biLSTM/CNN (Tan et al., 2016) network, and then the new neural network model with sentiment information, both metrics showed improvement on this test, with data composed of questions without sentiment.

The results for the proposed new neural network architecture and the baseline are as follows (statistically significant results are marked with '*'):

Table 10 Metrics for test on questions without sentiment

Metric	Precision@1	MRR
Baseline scores	0.4971	0.6881
New neural network scores	**0.5236***	**0.7041***
Improvement	2.64%	1.60%

For the new neural network, Paired T-Test is run to test for the statistical significance on questions without sentiment. The question-without-sentiment dataset is split into five equal parts, on which both the baseline and the new neural network architecture is run. The results are as recorded below:

Table 11 Precision@1 metric for questions without sentiment

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.5098	0.5315	0.5060	0.4655	0.4730
New neural network	0.5113	0.5405	0.5526	0.4895	0.5240

For the Precision@1 metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.02755, and the improvements seen are statistically significant.

Table 12 MRR metric for questions without sentiment

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.6973	0.7088	0.6967	0.6664	0.6719
New neural network	0.7021	0.7174	0.7196	0.6756	0.7063

For the MRR metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be .02221, and the improvements seen are statistically significant.

4.3.3 Discussion of results of two categories

Surprisingly, the performance for questions without sentiment is better than questions with sentiment for both the baseline and the new model, also the improvement rate is higher for questions without sentiment. The reason behind this scenario is analyzed as follows:

First, as described in section 4.2.1, the number of training samples belonging to the questions with sentiment category is 2626, while the number of training samples belonging to the questions without sentiment category is 18336. Their quantity ratio is close to 1: 6.98. Not enough training samples for the questions with sentiment is one possible reason why performance for this category is lower.

Second, the sentiment framework (Manning et al., 2014) we adopted for pruning the data into ‘questions with sentiment’ and ‘questions without sentiment’ is not the same as the sentiment framework (Felbo et al., 2017) we adopted for generating sentiment information for each question/answer. The difference in their design/judging criterion could lead to a different sentiment evaluation result.

Third, the quality of questions marked as ‘questions with sentiment’. If take positive questions in those questions marked as ‘with sentiment’ as an example, after examined through the positive questions, majority of questions are marked as ‘with sentiment’, because there are one or more words that are considered with ‘positive sentiment’ in the sentence, for example, “good,” “best,” “success,” whereas the sentence is actually a neutral question. Examples are provided in Figure 6. In this case, this sub-experiment can come with the conclusion that, Stanford Core NLP (Manning et al., 2014) could distinguish question with more sentiment than others, but this framework is not good enough to be used as the criterion for creating two independent test categories as ‘questions with sentiment’ or ‘questions without sentiment’.

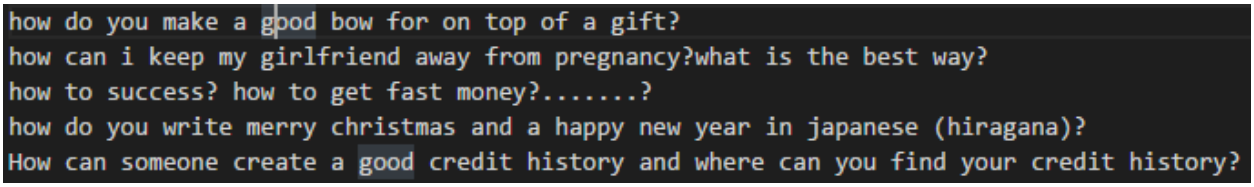

Figure 11 Examples of questions with sentiment belonging to the positive category

4.4 Sub-Test 2 subjective questions versus non-subjective questions

When pruning the dataset for subjective questions and non-subjective questions from the latter 70% of the entire dataset, the data for this sub-test is composed of 4000 subjective questions, and 4000 non-subjective questions, and their corresponding answers and sentiment information.

4.4.1 Results for subjective question

After running the baseline of the biLSTM/CNN (Tan et al., 2016) network, and then the new neural network model with sentiment information, both metrics showed improvement on this test, with data composed of subjective questions.

The results for the proposed new neural network architecture and the baseline are as follows (statistically significant results are marked with '*'):

Table 13 Metrics for test on subjective questions

Metric	Precision@1	MRR
Baseline scores	0.4457	0.6418
New neural network scores	**0.4581**	**0.6473**
Improvement	1.232%	0.55%

For the new neural network, Paired T-Test is run to test for the statistical significance on subjective questions. The subjective set is split into five equal parts, on which both the baseline and the new neural network architecture is run. The results are as recorded below:

Table 14 Precision@1 metric for subjective questions

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.4518	0.4397	0.4770	0.4382	0.4239
New neural network	0.4777	0.4397	0.4698	0.4468	0.4583

For the Precision@1 metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be .0944, and the improvements seen are not statistically significant.

Table 15 MRR metric for subjective questions

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.6472	0.6410	0.6621	0.6350	0.6256
New neural network	0.6658	0.6412	0.6577	0.6333	0.6405

For the MRR metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.15202, and the improvements seen are not statistically significant.

4.4.2 Results for non-subjective question

After running the baseline of the biLSTM/CNN (Tan et al., 2016) network, and then the new neural network model with sentiment information, both metrics showed improvement on this test, with data composed of non-subjective questions.

The results for the proposed new neural network architecture and the baseline are as follows (statistically significant results are marked with ‘*’):

Table 16 Metrics for test on non-subjective questions

Metric	Precision@1	MRR
Baseline scores	0.5042	0.6939
New neural network scores	**0.5319***	**0.7100***
Improvement	2.77%	1.61%

For the new neural network, Paired T-Test is run to test for the statistical significance on non-subjective questions. The non-subjective-question set is split into five equal parts, on which both the baseline and the new neural network architecture is run. The results are as recorded below:

Table 17 Precision@1 metric for non-subjective questions

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.5015	0.5501	0.4807	0.4931	0.4961
New neural network	0.4901	0.5840	0.5429	0.5331	0.5100

For the Precision@1 metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.04497, and the improvements seen are statistically significant.

Table 18 MRR metric for non-subjective questions

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.6947	0.7257	0.6770	0.6876	0.6855
New neural network	0.6917	0.7452	0.7142	0.7114	0.6886

For the MRR metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.04508, and the improvements seen are statistically significant.

In conclusion, the statistically significant matrixes are Precision@1, MRR. The statistically not significant matrix is None.

4.4.3 Discussion of results of two category

By comparing the performance between subjective questions and non-subjective questions, it is interesting to see that non-subjective questions category performs better than subjective questions for both the baseline and the new model. Also the improvement rate is higher for non-subjective questions. The reason behind this scenario is analyzed as follows:

First, the same case as questions with sentiment, there are not enough subjective questions for training. While the number of non-subjective questions is 13707, the number of subjective questions is 2429. Their quantity ratio is close to 5.64:1. Not enough training samples for subjective questions could be one possible reason why performance for this category is lower.

Second, since the methodology we adopted is a similarity-based method, it may not work well on subjective questions. For example, there is a subjective question like "How to find a perfect wife?" The answer to this question would mention personality, family background, hobbies, and more. This means the answer would have less overlap between words in the question and the answer, thus have a smaller similarity score. On the contrary, the non-subjective question would perform better since the answer would discuss in the same domain as the question proposed.

4.5 Sub-Test 3 non-subjective questions with sentiment versus non-subjective questions

When pruning the dataset for the non-subjective questions with sentiment and non-subjective questions without sentiment from the latter 70% of the entire dataset, the data for this sub-test is composed of 436 non-subjective questions with sentiment, and 4000 non-subjective questions, and their corresponding answers and sentiment information.

4.5.1 Results for the non-subjective questions with sentiment

After running the baseline of the biLSTM/CNN (Tan et al., 2016) network, and then the new neural network model with sentiment, both metrics showed improvement on this test, with data composed of the non-subjective questions with sentiment.

The results for the proposed new neural network architecture and the baseline are as follows (statistically significant results are marked with '*'):

Table 19 Metrics for test on non-subjective questions with sentiment

Metric	Precision@1	MRR
Baseline scores	0.4971	0.6881
New neural network scores	**0.5586**	**0.7315***
Improvement	6.15%	4.34%

For the new neural network, Paired T-Test is run to test for the statistical significance on non-subjective questions with sentiment. This set is split into five equal parts, on which both the baseline and the new neural network architecture is run. The results are as recorded below:

Table 20 Precision@1 metric for non-subjective questions with sentiment

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.4761	0.6094	0.4219	0.5000	0.4531
New neural network	0.4603	0.6719	0.6406	0.5313	0.4844

For the Precision@1 metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.08946, and the improvements seen are not statistically significant.

Table 21 MRR metric for non-subjective questions with sentiment

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.6743	0.7666	0.6514	0.6979	0.6514
New neural network	0.6786	0.7995	0.7753	0.7220	0.6874

For the MRR metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.04982, and the improvements seen are statistically significant.

4.5.2 Results for non-subjective questions

Results of non-subjective questions can be found in Table 6. Statistical significance and performance improvement are also analyzed in section 4.4.2.

4.5.3 Discussion of results of two category

For this sub-test, it gained the best improvement in the category of non-subjective question over other sub-tests.

The first reason could be a similarity-based method could gain better performance on non-subjective questions, because there would be more overlap between words since the answer would discuss in the same domain as the question proposed.

The second reason is that the new network can take advantage of the sentiment information in the non-subjective questions and their answers, and make a better prediction in the category of non-subjective questions with more sentiment contained. Although the Stanford core NLP could not to be used as the criterion for creating two independent test categories as ‘questions with sentiment’ or ‘questions without sentiment’ as previously addressed in section 4.3.3, it does have the ability to select questions with more sentiment contained within.

4.6 Sub-Test 4 subjective questions with sentiment versus subjective questions

When pruning the dataset for subjective questions with sentiment and subjective question from the latter 70% of the entire dataset, the data for this sub-test is composed of 511 subjective questions with sentiment, and 4000 subjective questions, and their corresponding answers and sentiment information.

4.6.1 Results for subjective questions with sentiment

After running the baseline of the biLSTM/CNN (Tan et al., 2016) network, and then the new neural network model with sentiment information, while Precision@1 saw improvement on this test, the MRR saw a minor drop in this sub-test, with data composed of subjective questions with sentiment.

The results for the proposed new neural network architecture and the baseline are as follows (statistically significant results are marked with ‘*’):

Table 22 Metrics for test on subjective questions with sentiment

Metric	Precision@1	MRR
Baseline scores	0.4266	0.6229
New neural network scores	**0.4289**	0.6217
Improvement	0.23%	-0.12%

For the new neural network, Paired T-Test is run to test for the statistical significance on subjective questions with sentiment. This dataset is split into five equal parts, on which both the baseline and the new neural network architecture is run. The results are as recorded below:

Table 23 Precision@1 metric for subjective questions with sentiment

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.4138	0.3523	0.5000	0.4659	0.4091
New neural network	0.4828	0.3750	0.4545	0.4432	0.3977

For the Precision@1 metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.45446, and the improvements seen are not statistically significant.

Table 24 MRR metric for subjective questions with sentiment

	Split 1	Split 2	Split 3	Split 4	Split 5
Baseline	0.6296	0.5701	0.6767	0.6350	0.6137
New neural network	0.6534	0.5948	0.6608	0.6267	0.5857

For the MRR metric, on running the Paired T test with the significance level set to 0.05, the p-value is calculated to be 0.47444, and the improvements seen are not statistically significant.

4.6.2 Results for subjective questions

Results of subjective questions can be found in Table 5. Statistical significance and performance improvement are also analyzed in section 4.4.1.

4.6.3 Discussion of results of two category

By referring to section 4.3.3, the previous experiment indicates that questions with sentiment do not perform well in its own category. Reasons include: Lack of data and different sentiment analysis frameworks lead to different design/judging criterion.

By referring to section 4.4.3, the previous experiment indicates that subjective questions do not perform well on its own category because the similarity-based method is not suitable for subjective questions, and also lack of data.

For this sub-test, the category of subjective questions with sentiment is a sub-set of subjective questions, it aims at the intersection of sentiment questions and subjective questions. So, it does make sense that combining those categories that did not perform well in its own category would not lead to an improvement in the end.

CHAPTER 5. CONCLUSION AND FUTURE WORK

In this study, we showed that adding sentiment information to the biLSTM/CNN (Tan et al., 2016) can improve the overall performance compared to the baseline using both Precision@1 and MRR evaluations. The Precision@1 performance gain reached 13.76% for the validation set, and 1.25% for the testing set. While the baseline model performance varied by 11.10% between validation set and testing set, the new model's performance is both higher and more stable, which indicates the new model has better ability to generalize to more data, and have a more consistent performance. As the performance of the new model shows improvement after adding sentiment information into the network, there should be inner relationships between sentiment information of question/answer and best answer, as selected by users.

In order to understand the performance better, four sub-tests were conducted using subjectivity and sentiment as criteria. For four sub-tests, the test for questions without sentiment shows more improvement over questions with sentiment. The test for non-subjective questions shows more improvement over subjective questions. The test for non-subjective questions with sentiment shows better improvement over non-subjective questions. The only sub-test showing negative performance difference is the test for subjective questions with sentiment, which MRR is 0.12% lower than subjective questions.

The proposed neural network architecture considering sentiment information does not give consistent performance in four sub-tests. Further investigation was conducted to look at the cause behind the performance difference between different categories of questions. Various potential reasons are summarized.

The potential reasons for why the performance various for different kinds of questions are summarized as follows:

- Not enough training data in one specific category of questions.
- Different sentiment analysis framework for generating the sentiment information, and categorizing questions into 'with sentiment'/ 'without sentiment'.

- Similarity-based method performs better for non-subjective questions.

5.1 Future work

Although this proposed network with sentiment could lead to an overall performance gain, there are several further improvements that could be done. Fixing each of the identified issues listed above can improve the results further.

First, if the dataset with a higher ratio of subjective questions or questions with sentiment could be found, the performance in those two categories could be improved since the model could extract more feature based on question-answer pairs in those specific categories.

Second, if sentiment analysis frameworks could give both sentiment information based on extended representation of text and also categorize them into with sentiment/without sentiment. This practice could avoid the problem of having different design/judging criterion between different sentiment analysis framework when performing sub-tests.

Third, further research could be done to investigate whether other Question-Answering approaches exists specifically for subjective questions. If found appropriate, a further experiment could be done adopting this kind of approach as the baseline and adding sentiment into the network, to verify its performance variance.

REFERENCES

Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. Proceedings of International conference of learning representations, 2015

Bastien, F., Lamblin, P., Pascanu, R., Bergstra, J., Goodfellow, I., Bergeron, A., ... & Bengio, Y. (2012). Theano: new features and speed improvements. arXiv preprint arXiv:1211.5590.

Cardie, C., Wiebe, J., Wilson, T., & Litman, D. J. (2003). Combining Low-Level and Summary Representations of Opinions for Multi-Perspective Question-Answering. In New directions in Question-Answering (pp. 20-27).

Chen, D., Bolton, J., & Manning, C. D. (2016). A thorough examination of the cnn/daily mail reading comprehension task. arXiv preprint arXiv:1606.02858.

Culpeper, J., Findlay, A., Cortese, B. & Thelwall, M. (2018). Measuring emotional temperatures in Shakespeare's drama. English Text Construction, 11(1), 10-37.

Eskandari, F., Shayestehmanesh, H., & Hashemi, S. (2015). Predicting best answer using sentiment analysis in community Question-Answering systems. In Signal Processing and Intelligent Systems Conference (SPIS), 2015(pp. 53-57). IEEE.

Felbo, B., Mislove, A., Søgaard, A., Rahwan, I., & Lehmann, S. (2017). Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. arXiv preprint arXiv:1708.00524.

Feng, M., Xiang, B., Glass, M. R., Wang, L., & Zhou, B. (2015). Applying deep learning to answer selection: A study and an open task. In 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU) (pp. 813-820). IEEE.

Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Networks, 18(5-6), 602-610.

Hasan, A., Moin, S., Karim, A., & Shamshirband, S. (2018). Machine learning-based sentiment analysis for twitter accounts. Mathematical and Computational Applications, 23(1), 11.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

Hochreiter, S., Bengio, Y., Frasconi, P., & Schmidhuber, J. (2001). Gradient flow in recurrent nets: the difficulty of learning long-term dependencies.

Ku, L., Liang, Y., & Chen, H. (2006). Opinion Extraction, Summarization and Tracking in News and Blog Corpora. In Proceedings of AAAI-2006 Spring Symposium on Computational Approaches to Analyzing Weblogs, AAAI Technical Report, 100-107.

Ku, L., Liang, Y., & Chen, H. (2007). Question Analysis and Answer Passage Retrieval for Opinion Question-Answering Systems. IJCLCLP, 13.

Kim, S. M., & Hovy, E. (2004). Determining the sentiment of opinions. In Proceedings of the 20th international conference on Computational Linguistics (p. 1367). Association for Computational Linguistics.

Li, C. N., & Thompson, S. A. (1989). Mandarin Chinese: A functional reference grammar. Univ of California Press.

Loria, S., Keen, P., Honnibal, M., Yankovsky, R., Karesh, D., & Dempsey, E. (2014). Textblob: simplified text processing. Secondary TextBlob: Simplified Text Processing.

Manning, C., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S., & McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations (pp. 55-60).

Mayo, M. (2017). A General Approach to Preprocessing Text Data. Retrieved from https://www.kdnuggets.com/2017/12/general-approach-preprocessing-text-data.html

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In Advances in neural information processing systems (pp. 3111-3119).

Mishra, A., & Jain, S. K. (2016). Computing Sentiment Polarity of Opinion WHY Type Question for Intention Mining of Questioners in Question-Answering Systems. Research in Computing Science, 110, 31-40.

Moghaddam, S., & Ester, M. (2011). AQA: aspect-based opinion Question-Answering. In 2011 11th IEEE International Conference on Data Mining Workshops (pp. 89-96). IEEE.

Nielsen, F. Å. (2011). A new ANEW: Evaluation of a word list for sentiment analysis in microblogs. arXiv preprint arXiv:1103.2903.

Oh, J. H., Torisawa, K., Hashimoto, C., Kawada, T., De Saeger, S., Kazama, J. I., & Wang, Y. (2012). Why Question-Answering using sentiment analysis and word classes. In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (pp. 368-378). Association for Computational Linguistics.

Pang, B., Lee, L., & Vaithyanathan, S. (2002, July). Thumbs up?: sentiment classification using machine learning techniques. In Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10 (pp. 79-86). Association for Computational Linguistics.

Quinlan, J. R. 2000. Data Mining Tools See5 and C5.0.

Rush, A. M., Chopra, S., & Weston, J. (2015). A neural attention model for abstractive sentence summarization. arXiv preprint arXiv:1509.00685.

Riloff, E., & Wiebe, J. (2003). Learning extraction patterns for subjective expressions. In Proceedings of the 2003 conference on Empirical methods in natural language processing.

Severyn, A., & Moschitti, A. (2015). Unitn: Training deep convolutional neural network for twitter sentiment classification. In Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015) (pp. 464-469).

Stack Exchange Data Dump: Stack Exchange, Inc.: Free Download, Borrow, and Streaming. (n.d.). Retrieved March 4, 2019, from http://archive.org/details/stackexchange

Stoyanov, V., Cardie, C., & Wiebe, J. (2005). Multi-perspective Question-Answering using the OpQA corpus. In Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing (pp. 923-930). Association for Computational Linguistics.

Somasundaran, S., Wilson, T., Wiebe, J., & Stoyanov, V. (2007). QA with Attitude: Exploiting Opinion Type Analysis for Improving Question-Answering in On-line Discussions and the News. In ICWSM.

Surdeanu, M., Ciaramita, M., & Zaragoza, H. (2008). Learning to rank answers on large online QA collections. Proceedings of ACL-08: HLT, 719-727.

Tan, M., Santos, C. D., Xiang, B., & Zhou, B. (2015). LSTM-based deep learning models for non-factoid answer selection. arXiv preprint arXiv:1511.04108.

Tang, D., Qin, B., & Liu, T. (2015). Document Modeling with Gated Recurrent Neural Network for Sentiment Classification. EMNLP.

Tran, N. K., & Niederée, C. (2018). A Neural Network-based Framework for Non-factoid Question-Answering. In Companion of the The Web Conference 2018 on The Web Conference 2018 (pp. 1979-1983). International World Wide Web Conferences Steering Committee.

Thom, J., & Scholer, F. (2007). A comparison of evaluation measures given how users perform on search tasks. In ADCS2007 Australasian Document Computing Symposium. RMIT University, School of Computer Science and Information Technology.

Turpin, A., & Scholer, F. (2006). User performance versus precision measures for simple search tasks. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 11-18). ACM.

Hatzivassiloglou, V., & McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In Proceedings of the 35th annual meeting of the association for computational linguistics and eighth conference of the european chapter of the association for computational linguistics (pp. 174-181). Association for Computational Linguistics.

Verberne, S., Boves, L., Oostdijk, N., & Coppen, P. A. (2008). Using syntactic information for improving why-Question-Answering. In Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1 (pp. 953-960). Association for Computational Linguistics.

Verberne, S., Boves, L., Oostdijk, N., & Coppen, P. A. (2010). What is not in the Bag of Words for Why-QA?. Computational Linguistics, 36(2), 229-245.

Verberne, S., van Halteren, H., Theijssen, D., Raaijmakers, S., & Boves, L. (2011). Learning to rank for why-Question-Answering. Information Retrieval, 14(2), 107-132.

Wang, D., & Nyberg, E. (2015). A long short-term memory model for answer sentence selection in Question-Answering. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers) (Vol. 2, pp. 707-712).

Wang, M., Smith, N. A., & Mitamura, T. (2007). What is the Jeopardy model? A quasi-synchronous grammar for QA. In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL).

Wiebe, J., Wilson, T., & Cardie, C. (2005). Annotating expressions of opinions and emotions in language. Language resources and evaluation, 39(2-3), 165-210.

Yang, L., Ai, Q., Spina, D., Chen, R. C., Pang, L., Croft, W. B., ... & Scholer, F. (2016). Beyond factoid QA: effective methods for non-factoid answer sentence retrieval. In European Conference on Information Retrieval (pp. 115-128). Springer, Cham.

Yao, X., Van Durme, B., Callison-Burch, C., & Clark, P. (2013). Answer extraction as sequence tagging with tree edit distance. In Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 858-867).

Zimbra, D., Abbasi, A., Zeng, D., & Chen, H. (2018). The State-of-the-Art in Twitter Sentiment Analysis: A Review and Benchmark Evaluation. ACM Transactions on Management Information Systems (TMIS), 9(2), 5.