ABSTRACT

ANALYZING FLUCTUATION OF TOPICS AND PUBLIC SENTIMENT THROUGH SOCIAL MEDIA DATA

by
Haoyue Liu

Over the past decade years, Internet users were expending rapidly in the world. They form various online social networks through such Internet platforms as Twitter, Facebook and Instagram. These platforms provide a fast way that helps their users receive and disseminate information and express personal opinions in virtual space. When dealing with massive and chaotic social media data, how to accurately determine what events or concepts users are discussing is an interesting and important problem.

This dissertation work mainly consists of two parts. First, this research pays attention to mining the hidden topics and user interest trend by analyzing real-world social media activities. Topic modeling and sentiment analysis methods are proposed to classify the social media posts into different sentiment classes and then discover the trend of sentiment based on different topics over time. The presented case study focuses on COVID-19 pandemic that started in 2019. A large amount of Twitter data is collected and used to discover the vaccine-related topics during the pre- and post-vaccine emergency use period. By using the proposed framework, 11 vaccine-related trend topics are discovered. Ultimately the discovered topics can be used to improve the readability of confusing messages about vaccines on social media and provide effective results to support policymakers in making their policy their informed decisions about public health. Second, using conventional topic models cannot deal with the sparsity problem of short text. A novel topic model, named Topic Noise based-Biterm Topic Model with FastText embeddings (TN-BTMF), is proposed to deal with this problem. Word co-occurrence patterns (i.e. biterms) are dirctly generated in BTM. A scoring method based on word co-occurrence and semantic similarity is proposed to detect noise biterms. In the

experiment part, the results demonstrate that the proposed TN-BTMF model outperforms its peers on three short text datasets.

ANALYZING FLUCTUATION OF TOPICS AND PUBLIC SENTIMENT THROUGH SOCIAL MEDIA DATA

by
Haoyue Liu

A Dissertation
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy in Computer Engineering

Helen and John C. Hartmann Department of
Electrical and Computer Engineering

August 2022

APPROVAL PAGE

ANALYZING FLUCTUATION OF TOPICS AND PUBLIC SENTIMENT THROUGH SOCIAL MEDIA DATA

Haoyue Liu

Dr. Mengchu Zhou, Dissertation Advisor — Date
Distinguished Professor, Department of Electrical and Computer Engineering, NJIT

Dr. John D. Carpinelli, Committee Member — Date
Professor, Department of Electrical and Computer Engineering, NJIT

Dr. Edwin Hou, Committee Member — Date
Professor, Department of Electrical and Computer Engineering, NJIT

Dr. Qing Liu, Committee Member — Date
Associate Professor, Department of Electrical and Computer Engineering, NJIT

Dr. Zhi Wei, Committee Member — Date
Professor, Department of Computer Science, NJIT

BIOGRAPHICAL SKETCH

Author: Haoyue Liu

Degree: Doctor of Philosophy

Date: August 2022

Undergraduate and Graduate Education:

- Doctor of Philosophy in Computer Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2022
- Master of Science in Computer Engineering,
  New Jersey Institute of Technology, Newark, NJ, 2017
- Bachelor of Science in Automation,
  Kunming University of Science and Technology, Kunming, P.R.China, 2014

Major: Computer Engineering

Presentations and Publications:

H. Liu, M. C. Zhou and Q. Liu, “An embedded feature selection method for imbalanced data classification,” *IEEE/CAA Journal of Automatica Sinica*, 6(3), 703-715, 2019.

H. Liu, I. Chatterjee, M. C. Zhou, X. S. Lu and A. Abusorrah, “Aspect-based sentiment analysis: A survey of deep learning methods,” *IEEE Trans. Computational Social Systems*, 7(6), 1358-1375, 2020.

H. Liu, M. C. Zhou, “Analysis of evolutionary social media activities: Pre and Post-vaccine Emergency Use,” *submitted to IEEE/CAA Journal of Automatica Sinica*, 2022, (under 2nd review).

H. Liu and M. C. Zhou, “Decision tree rule-based feature selection for large-scale imbalanced data,” in *Int. Conf. Wirless and Optical Communication Conference (WOCC)*, Newark, USA, April.7-8, pp. 1-6, 2017.

H. Liu, M. C. Zhou and X. S. Lu, “Weighted Gini index feature selection method for imbalanced data,” in *Int. Conf. on Networking, Sensing and Control (ICNSC)*, Zhuhai, China, March.27-29, pp. 1-6, 2018.

S. Han, K. Zhu, M. Zhou, X. Liu, H. Liu, Y. Al-Turki, and A. Abusorrah, ”A Novel Multiobjective Fireworks Algorithm and Its Applications to Imbalanced Distance Minimization Problems,” in *accepted by IEEE/CAA Journal of Automatica Sinica, 2022.*

X. S. Lu, M. Zhou, L. Qi and H. Liu, ”Clustering Algorithm-based Analysis of Rare Event Evolution via Social Media Data,” in *IEEE Trans. on Computational Social Systems*, 6(2), pp. 301-310, Apr. 2019.

X. S. Lu, M. Zhou, H. Liu and L. Q, ”A Comparative Study on Two Ground Truth Inference Algorithms based on Manually Labeled Social Media Data,” in *IEEE 16th International Conference on Networking, Sensing and Control (ICNSC)*, pp. 436-441. 2019

Dedicated to my family, all inclusive, known and unknown - for giving birth to me at the first place and supporting me spiritually throughout my life.

ACKNOWLEDGMENT

Foremost, I would like to express my deepest gratitude to my advisor, Dr. Mengchu Zhou, for his excellect guidance and patience. Professor Zhou not only served as my research supervisor, providing valuable and countless resources, insight, and intuition, but also constantly gave me support and encouragement.

I also give many thanks to Dr. Edwin Hou, Dr. John D. Carpinelli, Dr. Qing Liu and Dr. Zhi Wei for actively participating in my committee.

I have been given a wonderful opportunity as Teaching Assistant by the Department of Electrical and Computer Engineering and Research funding from Prof. Mengchu Zhou. Through these positions, I was able to better support my academic life and have better opportunities to interact with many excellect professors.

I would like to express my appreciation to the group members of Discrete Event Systems Laboratory: Dr. Xiaoyu Lu and Mr. Qi Kang with whom I have collaborated with my research, conducted insightful discussions, and undertook difficulties for many years. I also want to thank my good friends: Dr. Qi Liang, Dr. Xiwang Guo, Dr. Ziyan Zhao, Dr. Meiji Cui, Ms. Siya Yao, Ms. Chenwei Zhao, Mr. Hongjie Liu, and many others, who gave me strength and encouragement over the past five years.

I would like to thank my family, especially my parents, for support during my Ph.D. study.

TABLE OF CONTENTS

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

TABLE OF CONTENTS
(Continued)

Chapter **Page**

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES
(Continued)

Figure **Page**

CHAPTER 1

INTRODUCTION

Machine learning is an important branch of artificial intelligence, involving a number of disciplines such as statistics, matrix analysis, and optimization approaches. Its essence is to obtain general laws from data through automatic analysis, and use the learned general laws to make predictions on unknown data. It is also an important class of methods for data analysis in industry. With the rapid development of data collection and storage technology, a large amount of complex data has been accumulated. We have indeed entered the era of big data. In this era, machine learning provides reliable theoretical support and effective technical methods for intelligent data analysis, making it possible to mine useful knowledge from a large amount of structured and unstructured complex data.

In the era of big data, social media data is one of the main sources of big data. With their booming development, social networks have attracted a large number of users, and become one of the important ways for people to interact with each other in the current era. Take Facebook Inc. as an example. According to its 2021 financial report data, the company's main business, Facebook has 2.91 billion monthly active users [1]. WhatsApp has 2.3 billion monthly active users. Instagram has 2.1 billion monthly active users. The total number of all businesses is approximately 7.8 billion. Figure 1.1 shows the monthly active users of popular social media platforms. Especially after the prevalence of mobile intelligent terminals, more and more users can conveniently express their opinions, experience, follow the trend of hot events, and even participate in voting or making decision of important events on social media platforms anytime and anywhere. Twitter, one of the most popular social media platforms among Internet users generates hundreds of millions of messages on average per day. In addition, on the social media platform, both opinion leaders, traditional media users and grassroot civilians are the main sources of information about news events or social opinions, which may be related to politics, economy, people's livelihood, entertainment, technology and other aspects. This large user base has reinforced Twitter positions as a mainstream media. As a result, Twitter has become a valuable source with a huge amount of useful information, and has also proved to be a research resource for some important topic, such as, user interest mining, personalized recommendations, hot topic tracking, and public opinion monitoring.

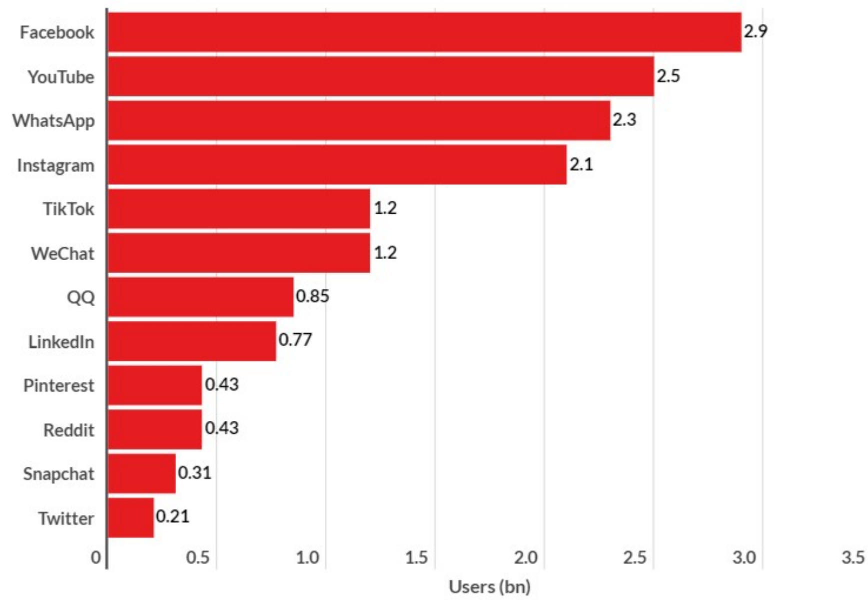

Figure 1.1 Monthly activity users of popular social media platform.

1.1 Sentiment Analysis

Sentiment analysis is a process of analyzing, processing, generalizing, and reasoning subjective texts with emotional overtones by using automated or semi-automated methods. It has been one of the most active research areas in natural language processing since the early 2000s. It mainly includes tasks such as classification, extraction, retrieval and summarization of sentiment information, and is often referred to as opinion mining [2].

The purpose of sentiment analysis is to extract ideas and sentiments from natural language texts. It brings the knowledge into a structured form for use and manipulation by decision makers. Sentiment analysis is a multidisciplinary research area. Erik *et al.* [3] mention that sentiment analysis combines artificial intelligence and semantic web techniques for knowledge representation and mining, mathematical knowledge for graph data mining and data dimensionality reduction, linguistic knowledge for semantic and pragmatic analysis, as well as sociological and psychological knowledge. It is a very challenging task for both industry and academia.

The basic task for sentiment analysis is to classify the expressed opinion of a given text into three classes which are positive, negative, and neutral. Searching for sentiment means finding a quadruple (S, G, H, T) [4], where S represents sentiment, G represents a target entity for which the sentiment is expressed, H represents the holder, i.e., the one expressing the sentiment, and T represents the time at which the sentiment is expressed. Note that most approaches focus only on finding the pair (S, G). In this case, the target is any characteristic or property of an entity. The target is made according to an application domain at hand.

In the public domain, social network sentiment analysis can be used for public opinion monitoring and event prediction. Public opinion refers to the sum of people's attitudes, thoughts and opinions generated by various social phenomena or issues over a period of time. Due to the convenience brought by the development of Internet technology, the right to speak is not in the hands of a few people, but in the hands of Internet users. Various issues and opinions related to the country's livelihood can be released at any time. At the moment a message is posted by an author, an online social network immediately recommends it to all followers of the author. It is then forwarded by the followers, which in turn spreads to more users. The information can be exploded in a short period of time, which may cause some social security issues. In addition, the sentiment analysis of social media data can be used to predict events. For example, during the 2012 election in the United States, scholars successfully predicted the election results by analyzing and mining the information on Twitter [5]. An important application of social media sentiment analysis in the business world is recommendation systems [6][7][8]. The key element to improve user experience is that users can set their personal preferences. By analyzing the sentiment in social media data, it is possible to accurately reflect the user's own preferences and can achieve the goal of recommending more accurate products and content to users. For example, by analyzing the sentiment of users' past movie reviews, it is possible to obtain the types of movies that users like in order to recommend similar movies to them. In addition, companies can analyze the content related to their products in social networks to obtain users' comments on various aspects of their products and collect user experiences, which can help them improve their products, gain a larger market share, and generate greater economic benefits.

1.2 Topic Modeling

In recent years, topic modeling analysis based on social media data has become a hot research topic in text mining in the social media field, and also one of the main ways to mine useful information from a huge amount of data. Since 1996, the research direction of topic identification and tracking learning has been gradually established under the promotion of related conferences and events. Research on topic mining has produced many mature models so far, and these models have additionally proved to be profitable in application areas such as topic identification of traditional texts. However, for the specific textual content structure of social media text, the traditional topic models have come to be no longer applicable. This is due to a number of characteristics that largely distinguish social media texts from traditional ones [9]:

1) Sparsity. The text length limit of 280 characters, which make the social media text shorter than the traditional one. This directly leads to the sparsity of social media texts, which makes it difficult for traditional topic models to extract useful features and explore the correlation among features.

2) Irregularity. The openness and interactivity of social media information for the general public make the text information more popular and the language simpler, and the wording less standardized and less strict. Due to the word limit of social media platforms and the lack of a terminology standardization system, more users choose to use abbreviations or buzzwords to express their opinions. All these highlight the irregularity of social media texts and bring many difficulties to social media text mining.

3) High dimensionality. Twitter generates hundreds of millions of texts per day and the content form of each texts is different, which brings some difficulties to the representation of texts. The general vectorized spatial representation of text is characterized by words (term/word), and thus a few thousand short texts may generate a vector of tens of thousands of dimensions. Such high dimensional vectors obviously increase the time consumption of their processing algorithms and lead to inefficiency.

4) Uneven distribution of topics. The information on social networks are often diverse and most of them are closely related to people's daily life, such as advertisements, weather and other daily topics, but also include issues related to users' personal mood, food, horoscope and so on. Often these messages are not the hot topics about which people really care, and are not conducive to the mining of useful information.

In order to address these problems, Phan *et al* [10] pre-trained topics on an external long-text dataset (e.g., Wikipedia) and then used these pre-trained topics to improve the effectiveness of topic modelling on short texts. Another strategy to address these problems is to aggregate short texts into pseudo-long texts based on certain feature of corpus, and then extract topics directly from the pseudo-long texts using standard topic models. Wang *et al* [11] first aggregated tweets from the same user into pseudo-long texts, and then directly used a Latent Dirichlet Allocation (LDA) model to model topics on the pseudo-long text and then analyze user characteristics. Similarly, it is possible to use other features such as timestamps and hashtags to aggregate short texts, thus improving the effectiveness of short text topic extraction.

1.3 Organization of Dissertation

This dissertation study aims to proposed an improved short text topic modeling method named Topic Noise based-Biterm Topic Model with FastText embedding (TN-BTMF). Our concerned dataset can be broadly classified into two types a) crawled from Twitter, consisting of COVID-19 vaccine related event; and b) publicly avaiable datset. The rest of the dissertation structure is as follows: Chapter 2 reviews relevant sentiment analysis and topic modeling work. Chapter 3 discusses a comparable study of aspect-based sentiment analysis, Chapter 4 analyzes the topic and sentiment trend in COVID-19 vaccine-related event, and Chapter 5 presents the proposed TN-BTMF models, and Chapter 6 showcases the conclusion and future work.

CHAPTER 2

LITERATURE REVIEW

2.1 Sentiment Analysis

Sentiment analysis has become a significant research direction in natural language processing (NLP). It consists of a combination of information retrieval, natural language processing, and artificial intelligence. Sentiment analysis is also known as opinion mining or subjectivity analysis. It studies various aspects such as opinions, sentiments, evaluations, appraisals, attitudes, and emotions [4]. The commonly used phrase for sentiment analysis is "opinion mining", which is derived from data mining and information retrieval community. Its main goal is to determine the opinions of a group of people on a certain topic. Sentiment analysis is a commonly used term that focuses on identifying the sentiment expressed in a text. It has become a rapidly growing research area since the year 2000 when Pang *et al.* [12] created a comprehensive study to determine sentiment polarity of movie reviews. It has received attention from not only academia but also industry because it can provide feedback information of customers through online reviews, help in deciding marketing policies, and detect changes in customer's opinions. Sentiment analysis is used to identify and extract opinions within texts, phrases, sentences, or documents. Its basic task is to classify the expressed opinion of a given text into three classes which are positive, negative, and neutral.

Sentiment analysis can be investigated at three levels:

1) Document-based sentiment analysis

In this task, sentiment analysis is used to determine if the overall document expresses a positive or negative opinion. Document-based is the simplest way of sentiment analysis. The task assumes that each document expresses an opinion on only one entity. Although, this assumption does not hold for cases where multiple entities are evaluated in one document. Therefore, a finer analysis is required.

2) Sentence-based sentiment analysis

In this level, the task is to determine if a sentence expresses a positive, negative, or neutral opinion about an entity. This level has a neutral opinion that does not exist in document-based sentiment analysis. A neutral opinion is the one in which no opinion is expressed in a sentence. It has the same assumption as document-based sentiment analysis, i.e., only one entity is expressed in a sentence.

3) Aspect-based sentiment analysis

The first two levels are very effective when an entire document or sentence points to a single entity. Nevertheless, people tend to talk about entities with many different aspects (attributes). For each aspect, people tend to have different opinions. Aspect-based sentiment analysis is the finest-grained analysis. This exists usually in product reviews from several online companies like Amazon, Yelp, and eBay regarding products such as cars, cameras, and mobile phones.

For example, in product reviews, a product itself is usually an entity, while all things related to it, e.g., price and quality, are its aspects. Aspect-based sentiment analysis aims to not only find the overall sentiment associated with an entity, but also the sentiment for each aspect of the referred entity. When a popular product has many reviews, potential customers have a hard time to read all the reviews before buying a product. In this case, a fine-grained analysis becomes very important. At the same time, it can provide manufacturers with more detailed information to help them improve their products in a specific aspect. For example, "Great Thai food but the service was dreadful". This comment reflects that the customer has a positive sentiment for 'food', but a very negative sentiment for 'service'. Hu *et al.* [13] believe that aspect-based sentiment analysis (ABSA) consists of two sub-tasks: 1) identifying aspects in which customers express their opinions; and 2) for each aspect, identifying sentences that give positive, negative or neutral opinions.

2.2 Topic Modeling for Short Text

With the profound development of Internet and explosively growth of online social media, short texts have become popular information carriers on Internet. A huge volume of short texts contain sophisticate information that can hardly be found in traditional methods. Discovering information hidden in large scale of short texts has been recognized as a challenging research problem.

Probabilistic topic models have been widely used, such as LDA, which assume a document is generated from a mixture of topics, where a topic is a probabilistic distribution of words. However, due to the limited length of short texts, much less co-occur words can detected than normal texts. Thus, a sparsity problem is founded, that caused the topic generated by these model is incoherent and elusive.

To alleviate the sparsity problem of implementing topic model on short text, several strategies are proposed. One straightforward strategy is to utilize plentiful auxiliary contextual information, such as hashtags, time, authorship, and locations to aggregate short texts into regular pseudo document [14][15]. Besides of the direct aggregation of short texts, Zuo *et al.* [16] propose a Pseudo-document-based Topic Model (PTM) for short texts without using auxiliary information. They assume each short text belongs to one and only one pseudo document. Another strategy, namely Dirichlet Multinomial Mixture (DMM) [17], restricts the text distribution over latent topics. It assumes that each text covers a single topic. The third strategy, namely Biterm Topic Model (BTM) [18], utilizes co-occurred word pairs mined from the corpus for topic inference.

All of the improvement strategies presented above only utilize the internal information of corpus while ignoring the external knowledge, such as the semantic similarity between two words. Word embeddings [19] are widely used in recent study. It aims at training a novel semantic representation of word from large scale of document. [20][21][22] incorporate word embeddings with topic model to tackle short texts sparsity problem.

CHAPTER 3

ASPECT-BASED SENTIMENT ANALYSIS

Sentiment analysis is a process of analyzing, processing, concluding, and inferencing subjective texts with the sentiment. According to the different needs for aspect granularity, it can be divided into a document, sentence, and aspect-based ones. This chapter summarizes the recently proposed methods to solve an aspect-based sentiment analysis problem. At present, there are three mainstream methods: lexicon-based, traditional machine learning, and deep learning methods. We provide their comparative review. Several commonly used benchmark datasets, evaluation metrics, and the performance of the existing methods are introduced.

3.1 Traditional Methods

ABSA is one of the fundamental tasks in the field of sentiment analysis. It consists of two main sub-tasks: aspect extraction and aspect-based sentiment classification [23] . Unlike document-based and sentence-based sentiment analysis, it considers sentiment and targets information at the same time. A target is usually an entity or aspect of an entity. The purpose of ABSA is to determine the sentiment polarity of a given sentence and aspect.

Based on the observation of Twitter sentiment, Jiang *et al.* [24] are the first ones to present the importance of targets and they also prove that 40% of classification errors in traditional classifiers are caused by their failure to consider the targets' information. They show that if traditional classifiers incorporate target-dependent features, they can gain better performance than target-dependent classifiers.

Kiritchenko *et al.* [25] propose a feature-based SVM for classification. It relies on the surface, lexicon, and parse features, and obtains acceptable performance in terms of accuracy. However, they cannot achieve very high performance, because it is restricted by sparseness and discreteness of features.

As a probabilistic generative model, Sentence Latent Dirichlet Allocation (SLDA) is proposed in [26]. It is used to solve the problem caused by using Latent Dirichlet Allocation (LDA) alone, i.e., LDA ignores the position information of words. An Aspect Sentiment Unification Model (ASUM), is proposed to extend SLDA. It incorporates both aspects and sentiment. Both SLDA and ASUM assume that words from one sentence are generated for a single topic.

Gupta *et al.* [27] introduce a feature selection technique for ABSA. The most relevant set of features for ABSA can be automatically extracted based on their proposed method. This method is based on Particle Swarm Optimization (PSO) [28], which is a computational method whose particles or solutions evolve iteratively till a local or global optimum is found. After removing the irrelevant set of features, Conditional Random Field (CRF) is used as a learning algorithm in [29] that can catch the most relevant features on the benchmark datasets of SemEval-2014 Task-4 [30].

The work [31] presents a novel context representation for ABSA, specifically for Twitter data. The traditional feature extraction methods are based on syntax. Nevertheless, the accuracy of using syntactic analysis is considerably lower on Twitter than on traditional text. This work uses a distributed word embedding and neural pooling function to enrich features automatically and solve the low parsing accuracy problem. But, this method highly depends on the effectiveness of the laborious feature engineering work and it can easily reach its performance bottleneck. Using a pooling function to capture syntactic and sentiment information for Twitter data is indeed over intuitive.

3.2 Deep Neural Network Model

With the rapid development of neural network methods in recent years, Deep Neural Networks (DNNs) have achieved great success in many applications. ABSA's research has shifted from feature engineering methods to deep neural network methods. Figure 3.1 shows the commonly used DNNs. They can be divided into 1) Convolutional Neural Network (CNN), 2) Recurrent Neural Network (RNN) [32], which include standard RNN, Long Short-Term Memory (LSTM) [33], and Gated Recurrent Unit (GRU) [34], 3) Recursive Neural Network (RecNN) [35], and 4) Memory Network (MN) [36]. In addition to the direct application of various DNNs and their variants, an attention mechanism combined with the above DNNs has become highly popular. Other types of ABSA methods include transfer learning and embedding methods. They are introduced in detail next.

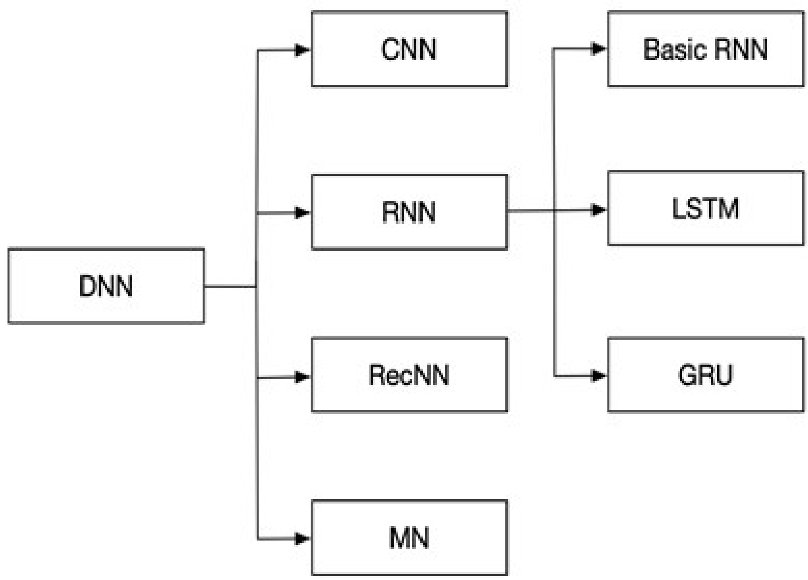

Figure 3.1 The categorization of Deep Neural Networks.

3.2.1 CNN based Methods

CNN helps image classification researchers achieve extensive breakthroughs. NLP researchers have applied CNNs to sentiment analysis, machine translation, and question answering, since 2011 when Collobert *et al.* [37] advocated CNN-based frameworks for NLP tasks. Figure 3.2 shows a basic CNN model [38]. Unlike the input of image processing, the input for NLP tasks is usually a sentence or a document represented by a matrix where each row represents a word and each word is represented by a vector. Using Figure. 3.2 as an example, it can be observed that the number of words in “I like this toy very much!” is 6, and the dimension of word embedding for these five words is chosen as 5. Hence, the input of this sentence is a matrix with dimension 6×5. The following CNN structure is owing to [38]. It comprises of an input layer, convolution layer with multiple filters, max-over-time pooling layer, and SoftMax classifier.

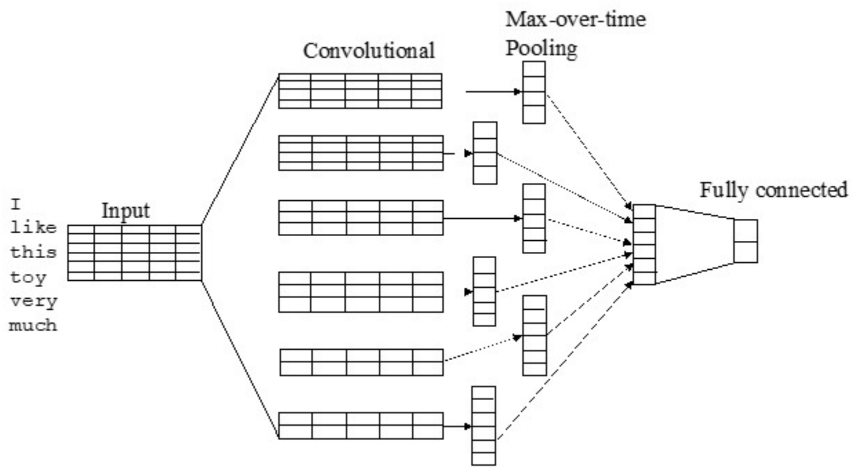

Figure 3.2 CNN model for sentence classification.

Input layer: An input layer is a matrix in which the order of word vector corresponds to the order of word in a given sentence. If a sentence has n-words and the dimension of word embeddings is k, then the matrix size is $n \times k$.

Convolutional layer: A convolutional layer is a result of moving a sliding-window over a sentence and then applying the same convolution filter to each window in a sequence. The size of a convolution window is $h \times k$, where h is the region size of a filter matrix. After the input matrix passes the convolution layer of an $h \times k$ convolution kernel, a feature map with one column is obtained. After the process of convolution is complete, a new feature c_i is obtained through an activation function, *e.g.* $tanh$. The value obtained by the convolution of adjacent h words of x_i, from x_i to x_{i+h-1}, is

$$c_i = s\left(w \cdot x_{i:i+h-1} + b\right) \tag{3.1}$$

where, s is a non-linear activation function, w is a convolutional filter, x_i denotes the i-th word in a sentence, and b is the bias term. Thus, in n n-word sentence, the result after the process of convolution is $c = [c_1, c_2, \cdots, c_{n-h+1}]$ which is called a convolution vector.

Pooling layer: A max-pooling layer method is adopted which combines the vectors resulting from different convolution windows into a single l-dimensional vector. This method is performed by extracting the maximum value observed in the previous vector from the convolutional layer.

Softmax layer: A one-dimensional vector is connected to the softmax layer for classification through the full connection.

3.2.2 RNN based Methods

RNN is a very popular model that has shown great power in many NLP tasks. The main idea behind it is to use sequential information. In traditional neural networks, we assume that all inputs are independent of each other. For many tasks, this is an unrealistic assumption. If one wants to predict the next word in a sequence, it needs to know which words are in front of it. RNN performs the same operations for each element in a series, relying on its previous calculations. In theory, RNN can use arbitrarily long sequenced information, but in reality, only a few previous steps can be reviewed. Since the number of input layers in a neural network is fixed, the variable length of input needs to process recurrently or recursively. The RNN realizes by dividing the variable length of input into some equal length of small pieces which are then inputted into the network. For example, when dealing with a sentence, the sentence is treated as a sequence of words. We then input one word at a time to RNN, until finishing the whole sentence. Finally, a corresponding output is produced through RNN. Figure 3.3 shows a basic RNN model. At time t, given x as input, we obtain the hidden state h_t:

$$h_t = \delta_h \left(U_h x_t + V_h h_{t-1} + b_h \right) \tag{3.2}$$

where, δ_h represents a logistic sigmoid activation function in the hidden layer, U and V are the weighted matrices for current input x_t, h_t, and h_{i-1} are the current and previous hidden state respectively, and b_h is the bias vector. After the hidden vector is obtained, output y_t is defined as:

$$y_t = \delta_{y(W_y h_t + b_h)} \tag{3.3}$$

where, δ_y represents a logistic sigmoid activation function and W_y is the weighted matrix for current input x_t.

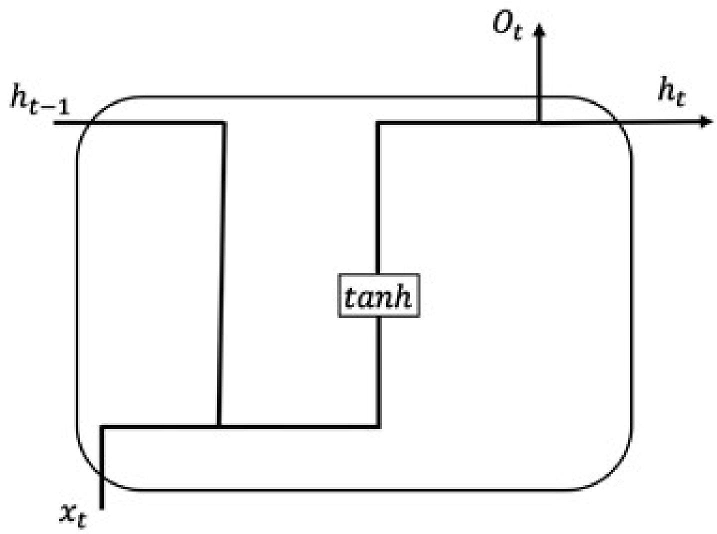

Figure 3.3 Basic Recurrent Neural Network (RNN) model.

Since the basic RNN faces a vanishing gradient problem, different functions are used to calculate the hidden states to solve this problem based on the sequence index position. The most common LSTM structure is shown in Figure 3.4. The key to LSTM's success is the cell state. The structures of the gates are used to remove or add information to a cell state. The first gate's structure is called forget gate, which is used to decide what information from the cell state will be discarded. The forget gate is calculated as follows:

$$f_t = \delta_f \left(W_f \left[h_{t-1}, x_t \right] + b_f \right) \tag{3.4}$$

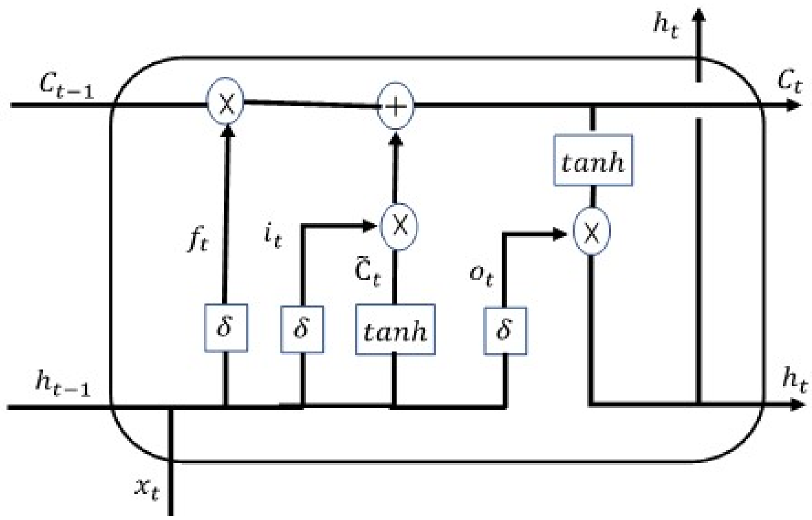

Figure 3.4 Long Short-Term Memory (LSTM) model.

where, δ_f represents a logistic sigmoid activation function, h_{t-1} is the previous hidden state, W_f and b_f are the weighted matrix and bias vector respectively for the current input x_t in a hidden state.

Then an input gate layer and a $tanh$ layer are used to decide what new information in cell state should be stored. Input i_t and candidate cell state $\tilde{C}_t$ are obtained as follows:

$$i_t = \delta \left(W_i \left[h_{t-1}, x_t \right] + b_i \right) \tag{3.5}$$

$$\tilde{C}_t = tanh \left(W_c \left[h_{t-1}, x_t \right] + b_C \right) \tag{3.6}$$

where, $tanh$ is a hyperbolic tangent function which helps in the rescaling of a logistic sigmoid where the output range of $tanh$ lies between -1 to 1.

After the input and new candiate cells are achieved, the new cell state C_t is updated with the help of the old cell state $C_{t=1}$:

$$C_t = f_t \bigodot C_{t-1} + i_t \bigodot \tilde{C}_t \tag{3.7}$$

where, $\bigodot$ denotes an element-wise function. After the new cell state is obtained, output o and hidden state h are calculated as follows:

$$o_t = \delta\left(W_o\left[h_{t-1}, x_t\right] + b_o\right) \tag{3.8}$$

$$h_t = o_t \bigodot tanh\left(C_t\right) \tag{3.9}$$

As a manifold network, GRU is a variant of an LSTM network. It is simpler and more effective than the original LSTM network. Like LSTM, it can solve the long dependency problem that troubles RNNs. It uses a hidden state to transfer information instead of using a cell state. It only contains two gates: reset and update gates. Figure 3.5 shows a basic GRU model, whose hidden state is calculated as follows:

$$r_t = \delta\left(W_r\left[h_{t-1}, x_t\right] + b_r\right) \tag{3.10}$$

$$z_t = \delta\left(W_z\left[h_{t-1}, x_t\right] + b_z\right) \tag{3.11}$$

$$\tilde{h_t} = tanh(W_h[r_t \bigodot \tilde{h_{t-1}}] + b_h) \tag{3.12}$$

$$h_t = (q - z_t) \bigodot h_{t-1} + z_t \bigodot \tilde{h_t} \tag{3.13}$$

where, r_t is a reset gate, which is used to decide the information to be eliminated. z_t represents an update gate, which helps in determining the information that needs to be passed along to the future. h_{t-1} and ht present the previous and currently hidden states respectively. The smaller the value of the reset state, the less information from the previous state is written to the current candidate set. The larger the value of the update gate, the more state information is brought into the current candidate set.

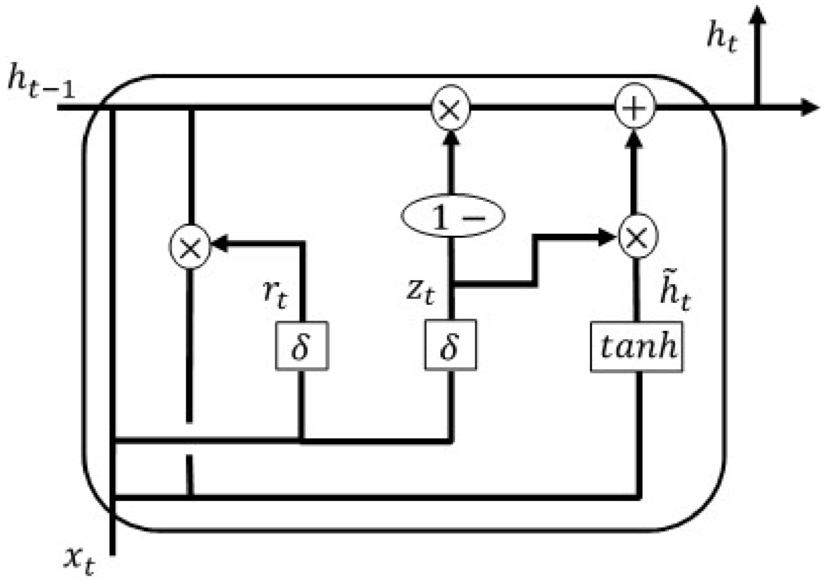

Figure 3.5 Gated Recurrent Unit (GRN) model.

3.2.3 RecNN-based Methods

RecNN is similar to RNN. Its computational graph is a deep tree, but it does not have the general RNN chain structure. Unlike RNN that can handle a fixed number of input layers, RecNN does not treat a sentence as a sequence of words. RecNN encodes the information, in the shape of a tree or a graph, as a vector and maps the information into a semantic vector space. This semantic vector space satisfies some kind of properties, for instance, vectors containing similar semantics are closer to each other in a space domain. In other words, if two sentences have the same meaning, their separately encoded vectors are close to each other.

The problem of RecNN is that its structure is a tree. Its computational time is ten times more than LSTM's. Putting a tree structure over a sentence means it needs to make categorical choices. It is used to determine the words that are going to be single components.

3.2.4 Memory Networks (MN)

The basic motivation of using Memory Networks is the need for long-term memory to hold the knowledge of questions and answers or the contextual information of conversations. A traditional RNN does not perform so well in long-term memory. Popular deep learning models such as, RNN, LSTM, and GRU use hidden states or an attention mechanism as their memory function, but the memory generated by them tends to be too small to accurately record all the information that is expressed in a paragraph. Thus, such pieces of information are lost while input is encoded as a "dense vector". Figure 3.6 shows a basic MN model which consists of memory m and four components: input feature map I, generalization G, output feature map O, and response R as follows [26]:

I: Given an input x, I convert x to internal feature representation $I(x)$.

G: Update old memories given the new input, m_i: $m_i = G(m_i, I(x), m)$.

O: Compute the output feature, o: $o = O(I(x), m)$.

R: Convert the output into the response format, r: $r = R(o)$.

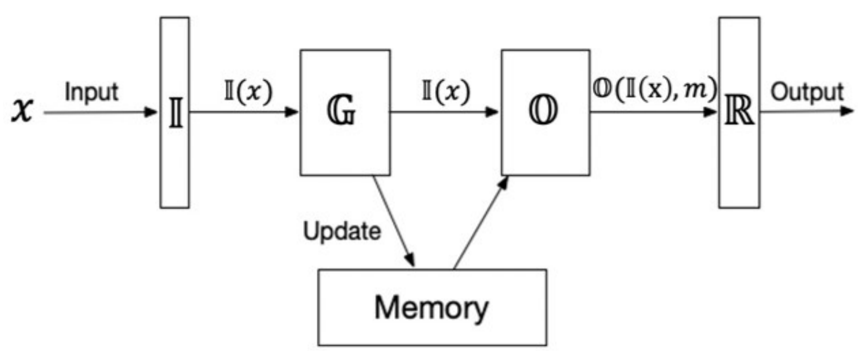

Figure 3.6 Memory Network (MN) model.

3.2.5 Encoder-Decoder with an Attention Model

There has been a growing interest in attention models for various applications, e.g., NLP, speech recognition, and computer vision. An attention model is primarily applied to image recognition. For example, it is used to imitate the focus of the gaze on different objects, when a person looks at an image. When a neural network is used to recognize a sentence, it gets more accurate if it only focuses on some specific features and ranks them accordingly. The most intuitive method to measure the importance of features is by giving weights to them. Therefore, an attention model is used to calculate such weights to distinguishes their importance.

In ABSA, it is important to model the intersection between an aspect and a sentence. A traditional encoder-decoder can encode irrelevant information, especially when an input sentence contains a huge amount of information. An attention mechanism is proposed in [39] to solve this drawback of an encoder-decoder structure. Figure 3.7 shows a basic encoder-decoder structure with an attention model [39]. There are different types of attention models used in ABSA. The detail is introduced in the following section.

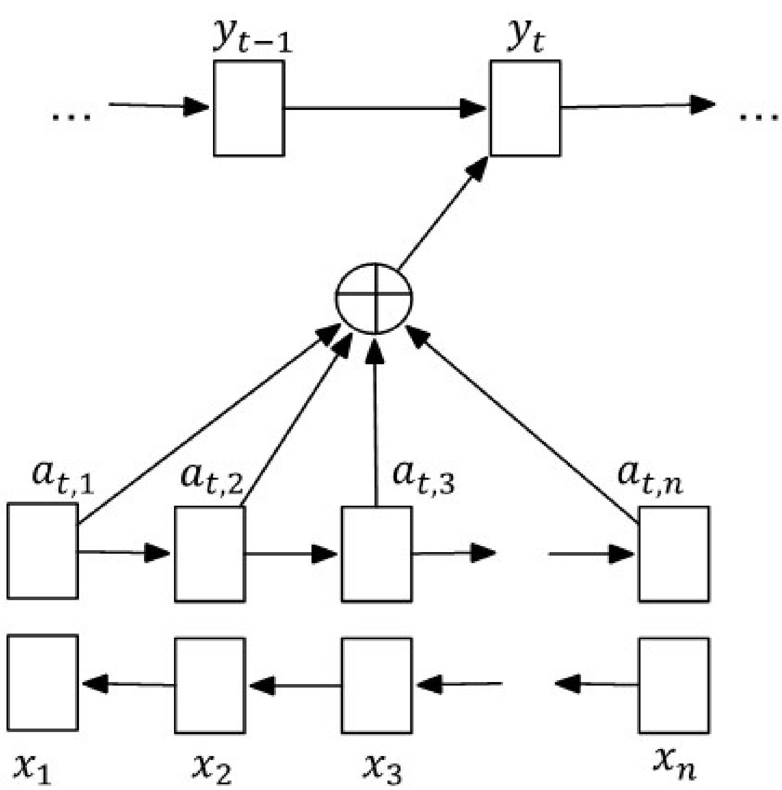

Figure 3.7 Encoder-Decoder architecture with an attention model.

3.3 Methodologies for ABSA

In this section, the existing methods are reviewed in detail according to the categories in Section 3.2.

3.3.1 CNN for ABSA

Some recent studies based on CNN models are summarized in Table 3.1. The table shows the proposed methods and reflects the critical ideas discussed in this chapter. The result comparison of these algorithms is shown in the section of Experimental Results.

Table 3.1 CNN-based Model in ABSA

Study	*Year*	*Method*	*Key Idea*
[40]	2018	PF-CNN PG-CNN	Implement CNN-based model for ABSA for the first time
[41]	2018	GCAE	Use gating mechanism to simplify the architecture of attention mechanism; Achieve highly-parallel during training process
[42]	2018	TNet-LF TNet-AS	Analyze the impact of attention weight based on noise interference and performance degradation
[43]	2019	IMN	Consider the joint information between aspect extraction and sentiment classification
[44]	2020	IGCN	Study the relation between a target and the context of a sentence or phrase in a review report

Two novel neural units proposed in [40] to fuse aspect information into CNN are Parameterized Filters for CNN (PF-CNN) and Parameterized Gated for CNN (PG-CNN). One drawback of standard CNN is its ignorance towards the information from aspect terms. PG-CNN overcomes this drawback by parameterizing filters using aspect terms. The former obtains the final classification feature by concatenating a targeted feature vector with a general one. PG-CNN adopts a parameterized gate to control the number of general features, which should pass to the final classification. A gate is formed from an aspect. Experimental results show that improvement in accuracy over standard CNN and some LSTM models.

CNN-based networks avoid independent modeling of targets for context-explicit representation. Kumar *et al.* [44] propose an interactive gate convolutional network (IGCN) that uses a bidirectional gating mechanism. Such a mechanism helps IGCN to learn the common relation between a target and the context of a sentence or phrase in a review report.

LSTM with attention mechanism has become popularly used structures for ABSA. However, their training process is very time-consuming. To solve this problem, Xue *et al.* [41] propose a Gated Convolutional network with Aspect Embedding (GCAE) model. This model is easier to be parallelized than LSTM. It works well for both aspect-category sentiment analysis (ACSA) and aspect-term sentiment analysis (ATSA). Its main components are convolution and gating mechanisms. For ACSA, word embeddings are obtained through two processes. The word embeddings combined with two separated convolutional layers, and then the two outputs and aspects are embedded in the proposed novel gating unit. The units include a tanh gate and ReLU one. In ATSA, one convolutional layer for target representation is an extended ACSA model to include another convolutional layer for the target expressions.

When using an attention mechanism, the attention weight may be subject to noise interference and performance degradation. To address these two issues, Li *et al.* [42] propose a target-specific transformation network (TNet). The contextualized word representation is obtained from a Bi-LSTM layer. First, contextualized word representations are transferred and generated by encoding context information into word embedding with a Bi-LSTM. Then, a target-specific transformation component is placed to incorporate the target information into contextualized word representations. Finally, a position-aware convolutional layer is designed to extract the relevant sentiment features of a given target.

An ABSA task is divided into two subtasks: aspect extraction and sentiment polarity classification. Most methods perform them in a pipeline order. However, if a pipeline order is followed, the joint information from them cannot be fully utilized. It causes the correlation between aspect extraction and sentiment classification not explicitly modeled. He *et al.* [43] propose an interactive multi-task learning network (IMN) that can learn aspect extraction and sentiment polarity classification simultaneously. A message-passing architecture is introduced in IMN, where information is passed to different tasks through a shared set of latent variables.

In general, CNNs are effective and can extract meaningful local patterns (n-grams) because they can mine semantics in contextual windows. However, it is difficult to maintain sequential order and model long-distance contextual information. RNN models are discussed next, to better solve this problem.

3.3.2 RNN for ABSA

Recent studies have majorly focused on three types of RNN models which are summarized in Table 3.2 - 3.3. Using a pooling function has two disadvantages; firstly, such usage cannot allow one to select useful features like syntactic and semantic information from Tweeter data. Such features are only valid when facing a sequence of words. Secondly, its use cannot allow one to display explicit information about the interaction between a target and its context. In response to these two shortcomings, Zhang *et al.* [45] propose a gated recurrent neural network (GRNN) based on [31]. Note that the method in [31] cannot accurately extract the latent semantic information e.g. dependency relations, co-references, and negative scopes. Also, a three-way gate (G3) is introduced to model the interaction between the target and its surrounding context in GRNN.

Cheng *et al.* [46] propose a hierarchical attention network to learn the aspect of information and aspect-specific sentiment information. A location mask layer is added to get the location information of aspect terms. Their proposed architecture is called

Table 3.2 RNN-based Model in ABSA

Study	*Year*	*Method*	*Key Idea*
[45]	2016	GRNN	1) Extract syntactic and semantic information from Twitter Data; 2) Display explicit information about the interaction between a target and its context
[46]	2017	HEAT	1) Consider the aspect-related information in the text; 2) Propose a hierarchical attention model: aspect attention and sentiment attention
[47]	2016	TD-LSTM TC-LSTM	Integrate the connections between target word and context word for the first time
[48]	2016	H-LSTM	1) Consider both inter- and intra-sentence relations; 2) Elaborate tasks for languages that lack large corpora and manually crafted linguistic resources
[49]	2016	AE-LSTM AT-LSTM ATAE-LSTM	Explore the connection between an aspect and the content of a sentence in ABSA using attention mechanism
[50]	2017	BILSTM-ATT-G	1) Learn the sentence split method in [**?**]; 2)Propose an attention mechanism to calculate the contribution of each word towards a targeted sentiment class
[51]	2017	IAN	Consider a target containing multiple word
[52]	2018	MGAN	1) Propose a fine-grained attention mechanism; 2) Learn attention weights by adopting a position encoding mechanism instead of simply averaging the aspect and context vectors
[40]	2018	AOA-LSTM	Solve the ignoring word-pair interaction information problem in [51]

Table 3.3 RNN-based Model in ABSA

Study	*Year*	*Method*	*Key Idea*
[53]	2017	AF-LSTM	Avoid training difficulty in [49]
[54]	2018	PRET-MULT	1) Consider insufficiently annotation problem; 2) Incorporate document-level labeled data for ABSA for the first time
[55]	2018	Inter-aspect dependencies	Learn the inter-aspect dependencies between aspect and their contextual information
[54]	2018	LSTM + SynATT + TarRep	Improve the ability of capturing a sematic information from a complex expressions' targets
[56]	2019	Soft label strategy	Avoid the noise caused by computing attention weights for word-level features
[57]	2019	CAN	Propose a module to simultaneously detect the sentiment for multiple aspects in a single sentence at the same time
[58]	2019	ATLS	Avoid the over-fitting problem caused by the sparse of attention vector

HiErarchical ATtention (HEAT) which consists of three parts: an input module for aspect and text embedding, a hierarchical attention module, and a sentiment classification module. Its text embedding is achieved by entering word embeddings into a bidirectional gated current unit (BiGRU). Its second module mines aspect information from aspect attention, and then uses this information to help HEAT grab sentiment information.

Except for using a gated recurrent neural network, there are a large number of studies based on the LSTM model. Tang *et al.* [59] introduce two target-dependent LSTM models. Their main idea behind the model is to integrate target information into LSTM. The model is trained in an end-to-end way with a standard backpropagation algorithm [60]. The first one is target-dependent LSTM (TD-LSTM), which is a fine-tuned LSTM. It uses two LSTM models to describe the left and right sides of a target's surrounding contexts plus target strings, respectively. They claim that if a target string is placed in the last unit, the semantics of the target can be better represented. Because TD-LSTM cannot capture the interaction between contexts and a given target, Tang *et al.* [48] propose a target-connection LSTM (TC-LSTM), which advances TD-LSTM. It extends a target connection component that can specifically represent the interaction information between a target and context words. According to their experimental results, TC-LSTM outperforms both TD-LSTM and LSTM.

The work in [48] proposes a hierarchical bidirectional LSTM (H-LSTM) that uses the relation between an intra-sentence and inter-sentence. It helps in avoiding the drawbacks of the previous neural network-based architectures that only considers the relation in an intra-sentence. It is suitable for languages that lack large corpora and manually crafted linguistic resources because, in H-LSTM feature engineering, positional information, and parser tree are not required. H-LSTM uses two Bi-LSTM models for sentence-level and review-level sentiment analysis, respectively. In the beginning, word embeddings are delivered into the Bi-LSTM [61] for sentence-level sentiment analysis, and then the output of both forward and backward LSTM is concatenated to the input of a review-level Bi-LSTM. Finally, the last layer of predicted sentiment is obtained by cascading review-level forward and backward LSTM outputs.

Even though content information plays an important role in deciding sentiment polarity, but it is not the only useful feature. Wang *et al.* [49] propose an attention-based LSTM with Aspect Embedding (ATAE-LSTM) model to explore the connection between an aspect and the content of a sentence. They utilize an attention mechanism to enforce a given aspect, correctly focusing on its highest related part in a sentence. Since aspect-specific sentiment information is directly used in an attention mechanism, an unrelated sentiment word may not match any aspect even if it contains semantic meanings in an aspect. This improves the accuracy by 3% as compared to LSTM.

Liu *et al.* [50] proposes another work by using an attention model on ABSA. The model is based on target-dependent features [31] and GRNN [45]. Here a sentence is split into three parts: targets, targets left contexts, and targets right contexts. A Bi-LSTM is adapted to extract word embeddings, and then an attention model is applied to the hidden nodes to measure the importance of each word. Experimental results show that their model achieves some improvement over the methods mentioned in [31] and [45].

All introduced models so far only consider the situation that contexts are formed from many words. However, a target is not limited to only one word. It means that even though the interaction between contexts and targets is the one to be learned, the vice versa can be modeled separately. According to this idea, Ma *et al.* propose an interactive attention network (IAN) to learn the interaction between contexts and targets and generate their representations separately [51] . The model is based on an LSTM with an attention mechanism. The targets and context word embeddings are fed into an LSTM. Two attention mechanisms are used to select the important information from the representations of initial contexts and targets, respectively, and then a new representation of contexts and targets are formed.

The traditional attention mechanisms learn attention weights by simply averaging the aspect and context vectors, which is at the coarse-grained level. When an aspect contains multiple words or a larger context, the information loss is caused by simply averaging aspect and context vectors. Also, previous methods fail to consider the situation that aspects contain multiple instances. Hence, Fan [52] proposes a novel multi-grained attention network (MGAN) to overcome these two drawbacks. Its architecture consists of an input embedding layer, contextual layer, multi-grained attention layer, and output layer. Aspect and sentence contextual output is acquired by feeding the aspect and context embedding into a Bi-LSTM, respectively. The closer a context words towards an aspect, the greater is the impact. Therefore, they adopt a position encoding mechanism to process the contexts. The fine-grained attention mechanism links and fuses information from the aspect and context words. The multi-grained attention layer is formed by concatenating both fine-grained and coarse-grained attention vectors.

When using the pooling operation in IAN Ma *et al.* [51], the word-pairs interaction information, which forms between sentences and targets, is ignored. Huang *et al.* [40] propose an attention-over-attention (AOA) neural network, which is an advanced version of IAN, to solve the ignoring interaction information problem. IAN feeds the word vectors of aspects and sentences into two Bi-LSTMs, respectively. These Bi-LSTMs enable two hidden semantic representations and then they use an AOA to calculate attention weights. Finally, semantic polarity is predicted by combining attention weights and implicit semantics. However, it is impossible to accurately classify sentiment polarity while facing a complex sentiment expression.

ATAE-LSTM [49] intends to answer whether concatenating of aspect and word at both LSTM and attention layers is needed. Although training becomes difficult when concatenation of aspect and words are added to both LSTM and attention layer. To avoid this difficulty, Tay *et al.* [62] introduce an aspect-fusion LSTM (AF-LSTM), in which a novel word-aspect fusion is proposed for the attention layer. Two association memory operators, circular correlation, and circular convolution are used to model the relationship between aspect embedding and context words. The proposed model outperforms ATAE-LSTM by 2% - 3% in accuracy.

The final goal of ABSA is to correctly analyze the sentiment for each aspect. To achieve high accuracy in sentiment analysis, annotation is one of the major problems. As DNN models play an essential role in ABSA, efficient annotation of aspect data becomes a crucial problem, because it directly affects the performance of neural network models. While there are many insufficiently labeled data for ABSA, a large amount of document-level labeled data can be provided through online reviews such as Yelp or Amazon. He *et al.* [54] propose a method to transfer knowledge from a document-level to a sentiment-level to solve this sub-problem. The work extends attention-based LSTM with pre-training and multi-task learning approaches to transfer document-level knowledge to improve the performance of ABSA. [54] is one of the first works to incorporate knowledge from the document-level corpus for the performance improvement of ABSA where insufficient sentence-level labeled data must be handled.

A large number of inter-aspect dependencies between aspect and their contextual information, such as incomplete information, and sentiment influence in conjunctions, are ignored in many current neural-based models. The work [55] propose an inter-aspect dependency model for ABSA. It contains two phases: 1) An attention-based LSTM model which is used to obtain aspect-based sentential representations; 2) An LSTM model which is used to capture the relationship among the aspects.

A novel method is proposed in [63] to improve the effectiveness of an attention mechanism. The work is based on an attention-based LSTM model. A new target representation method is introduced to better capture the aspect semantics of a given target in this model. The previous target representation is achieved by averaging the component word vectors of a target. When the expression of a target becomes more complex, it is not able to successfully receive its related semantics. The work [63] studies both aspect embeddings and target representation based on an autoencoder structure. It proposes an attention model for integrating syntactic information into an attention mechanism. It then uses a dependency parser to get syntactic information. The experimental results show that this method [63] can achieve a better performance than the conventional attention-based LSTM by 3%-6% increase in accuracy.

As DNNs receive a superb performance on ABSA, the combination of attention mechanism and DNN models have received extensive attention. However, when using an attention mechanism to calculate word-level features, some noise may be introduced in the model. Yin *et al.* [56] propose a soft label approach instead of using an attention mechanism, that consists of four parts which are Bi-LSTM layer, convolutional layer, LSTM layer, and sentiment classification, a context-aware representation is obtained at a Bi-LSTM layer. The local active feature is then captured at a convolutional layer. Deeper interaction between the context and a target is comprehended at the LSTM layer. Finally, the soft labels and positional weights are obtained and used for sentiment classification.

Hu *et al.* [57] studies that most of the sentences contain multiple aspects. 85% of multiple aspect sentences are non-overlapping. One of the problems is to simultaneously detect sentiment for multiple aspects in a single sentence because only a few words connect to the sentiment information in each aspect. The other problem is that the attention weight matrix of all aspects is sparse. Constrained attention networks (CAN) is proposed to solve these two problems. The model introduces two constraints that are sparse and orthogonal regularization on attention weights. Experimental results show that applying CAN on ATAE-LSTM increases accuracy by 5.39% and the F1 score by 6.46%.

Bao *et al.* [58] propose attention and lexicon regularized LSTM (ATLS), which is based on AT-LSTM, to improve an end-to-end learning system's performance. Also, previous study uses an attention vector sparser that may, unfortunately, cause an over-fitting problem. ATLS is proposed to solve the low resources and over-fitting problems.

3.3.3 RecNN for ABSA

The computational graph for RecNN is a deep tree, unlike RNN that has a chain structure. Some recent studies have focused on RecNN models which are summarized in Table 3.4.

Table 3.4 RecNN-based Model in ABSA

Study	*Year*	*Method*	*Key Idea*
[64]	2014	AdaRNN	Select composition functions in automated way
[22]	2015	PhraseRNN	Solve the shortcoming in [64] by assigning different weights to each context words, which differently contribute to a given aspect
[65]	2016	RNCRF	Propose a joint model for explicit aspect and opinion terms con-extraction

An Adaptive RNN (AdaRNN) is proposed for Twitter ABSA in [64]. AdaRNN is used to automatically select the composition functions instead of choosing them by hand-crafted rules. This method treats context words as being equally important given multiple aspects. However, in most of the cases, only a few context words are highly related to the sentiment polarity for a given aspect. Therefore, it is difficult to achieve good performance in practice. Nguyen *et al.* [22] introduce an extended model of RecNN and AdaRNN, which is called phrase recursive neural network (Phrase RNN). It assigns different weights to each context words and these words contain different contributions towards a given aspect. Phrase RNN extends AdaRNN by using two composition functions in inner- and outer-phrase. The AdaRNN is directly using a list of global functions. So, it improves AdaRNN by 8.7% accuracy.

To extract the term of both aspects and opinions, Wang *et al.* [65] introduce a novel joint model that combines a RecNN and condition random fields, namely recursive neural conditional random fields (RNCRF). High-level feature representation is learned through a dependency-tree recursive neural network (DT-RNN). Then, the output of DT-RNN is fed into a linear-chain conditional random field. A discriminative mapping from high-level features to labels is gained through conditional random fields.

3.3.4 Memory Network for ABSA

A Memory network (MN) is a method that needs a long-term memory to maintain the contextual information of a conversation. It has been wildly used in ABSA in recent studies. Some MN models are summarized and shown in Table 3.5.

Table 3.5 MN-based Model in ABSA

Study	*Year*	*Method*	*Key Idea*
[47]	2016	MemNet	Achieve a lower computational time than RNN models
[53]	2017	DyMemNN	1) Propose a novel extension of [47]; 2) Solve the shortcomings in standard MNs
[66]	2017	RAM	1) Propose method which is more robust against irrelevant information; 2) Adopt Bi-LSTM to produce the memory for the first time
[32]	2018	MN+GRU	Solve the lower performance problem when a context word's sentiment is sensitive to a given target in MNs
[67]	2018	Cabse	1) Consider partial context information for a given aspect, and ignored the irrelative ones; 2) Consider the ironical and sarcastic statements; 3) Consider multiple aspects in a sentence
[68]	2019	FCMN	Enrich the input resources of context words

Tang *et al.* [47] introduce a deep MN with an attention mechanism model for ABSA. The model has a lower computational time as compared to all RNN models. Several attention mechanisms are used to leverage both content and location information. Also, location attention has been added to receive location information between context words and aspects.

Two kinds of dynamic MNs are proposed in [53] which are Tensor DyMemNN and Holo DyMemNN. They represent a novel extension of MemNN [69]. They are used to address the shortcomings of using standard MNs [69] that represent the interaction between the aspect and words through dot products and feed-forward neural networks. The main motivation of this work is to achieve richer interaction between the aspect and words to enhance the learning capabilities. The proposed model is superior to the baseline MemNN.

A novel framework based on neural networks is proposed in [66] to identify the sentiment of opinion targets. It consists of an input module, memory module, position-weighted memory module, recurrent attention module, and output module. Using Bi-LSTM as the memory module, the output can synthesize the word sequence features. On this basis, the important context information is picked by using multiple attentions on MN.

The work in [32] proposes to combine multiple MNs with a delayed memory update mechanism to achieve the capability of tracking and updating the states of entities. Its structure for the delayed memory update is based on a gating mechanism, that was proposed in [34]. The delayed memory control is used when a gate is turned on and a past memory influences the current one. ABSA performance of the MNs model is lower when a context word's sentiment is sensitive to a given target. The work [70] proposes a multiple-target sensitive MNs (TMNs) to solve this problem. Their usage can capture the interaction between a target and its context words.

Through the analysis of the existing neural attention models mainly three problems are observed. First, their attention mechanism only considers partial context information for a given aspect. Yet some sentimental words are irrelative to the aspect. Second, the overall meaning of a sentence is ignored. When analyzing a complex sentence, ironical or sarcastic statements are easily ignored. Third, within a given topic multiple aspects may be contained. Three attention mechanisms are proposed in [67]. To solve the first two problems, a sentence-level content attention mechanism (SAM) is proposed.

Aiming at solving the last problem, a context attention-based memory module (CAM) is proposed. Based on both SAM and CAM, a content attention-based aspect-based sentiment classification model (Cabasc) is introduced. Their model achieves highly competitive performance on the dataset SenEval 2014.

The external memory only contains the word embedding information in MNs, which is not conducive to ABSA. Enriching the input resources of context words is a main issue to be addressed. The work [68] proposes a feature-based compositing memory networks (FCMN) to do so. It uses compositing strategies to combine aspect embedding, context embedding information, and multi-angle features as new input representation. The multi-angle features contain location, part-of-speech (POS), and sentiment features. There are three types of compositing strategies: front-compositing, inside-compositing, and rear-compositing which are used to gain context representations by combining the context features and context embedding. The performance of FCMN achieves a 10% improvement in accuracy than basic LSTM.

3.4 Datasets and Evaluation Results

3.4.1 Datasets

In this section, some publicly available datasets are introduced in detail. Table 3.6 summarizes some commonly used datasets and their statistics. SemEval 2014, SemEval 2015 [30] and SemEval 2016 [71] are the three most popular benchmark datasets, which are publicized by the international workshops on semantic evaluation. Also, the Twitter dataset released in [64] and SentiHood dataset released in [72] are frequently used for ABSA as well. Mitchell [73] and MPQA [74] are two datasets that are rarely used in recent researches.

Table 3.6 Statistic of Public Datasets

Dataset	*Domain*	*Size*	*Positive/Negative/Neutral*
SemEval 2014 [30]	Restaurants; laptops	Train: 3041, Test: 800; Train: 3045, Test: 800	987/866/460; 341/128/169; 2164/805/3693; 728/196/196
SemEval 2015 [30]	Restaurants; laptops; hotels	Train: 1654, Test: 845; Train: 1974, Test: 949; Train: NA, Test: 339	72.43%/24.36%/3.20%/; 53.72%/40.96%/5.32%/; 55.87%/38.75%/5.36%/; 57%/34.66%/8.32%/; 71.68%/24.77%/3.53%
SemEval 2016 [71]	Restaurants; laptops;	Train: 350, Test:90; Train: 450, Test: 80	
Twitter [64]	Twitter	Train: 6248, Test: 692	25%/ 25%/ 50%
SentiHood [72]	urban; neighborhoods	Single location: 1353; Two location: 1353	
Mitchell [73]	Twitter	Spanish: 30,000; English: 10,000	
MPQA [74]	Opinions and other private states	70 documents	

3.4.2 Evaluation Results

Table 3.7 -3.8 summarizes the performance of accuracy (α) and macro-F1 (M^F) for the SemEval 2014 dataset on the restaurant domain. The accuracy of the basic CNN model is 77.95%. The performance of IMN is α= 83.89% and M^F = 75.66%, which is highest for all the CNN models. This has the best performance in Table VII with 7% increase in accuracy. It can be inferred that the joint information, which is extracted from aspect extraction and sentiment classification, is one of the useful features to improve the performance. Adding document-level labeled corpora is an effective way to increase the training information. Even though GCAE does not obtain a good performance, this method concentrates on solving the time-consuming problem, which is a big challenge when using a deep learning model. GCAE has similar results as ATAE-LSTM, but it saves 87% of run time.

The accuracy and macro-F1 for CAN are 83.33% and 73.23% respectively which is the best performance in all the RNN models as can be seen in Table 3.8. CAN improve the performance by mainly focusing on solving the non-overlapping multiple aspects and sparse regularization problems. Table 3.9 - 3.10 summarizes the performance of accuracy and macro-F1 for the SemEval 2014 dataset from the laptop domain. In this table, both the TNet models contain better accuracy than IMN for the CNN model. Also, TNet-AS has the highest accuracy among all the models although, IMN has the best performance of M^F. The highest performance of accuracy and M^F in the RNN model is MGAN. Except for CAN, MGAN has the best performance in Table 3.8.

Comparing with SemEval 2014 dataset, only a small number of researchers work with a Twitter public dataset. The comparison results for this dataset are shown in Table 3.11. Two TNet models still obtain the best accuracy and M^F values.

Table 3.7 Comparison Study for SEMEVAL 2014 Dataset on Laptop Domain

Method	*Accuracy*	*Macro-F1*
CNN	77.95%	-
PF-CNN	79.20%	
PG-CNN	78.93%	-
GCAE	77.28%	-
TNet-LF	80.79%	70.84%
TNet-AS	80.69%	71.27%
IMN	83.89%	75.66%
TD-LSTM	75.60%	-
AE-LSTM	76.60%	66.45%
AT-LSTM	76.20%	-
ATAE-LSTM	77.20%	65.41%
BILSTM-ATT-G	79.73%	69.25%
IAN	78.6%	-
MGAN	81.25%	71.94%
AOA-LSTM	81.20%	-

Table 3.8 Comparison Study for SEMEVAL 2014 Dataset on Restaurant Domain

Method	*Accuracy*	*Macro-F1*
AF-LSTM	75.40%	-
PRET+MULT	79.11%	69.73
Inter-aspect dependencies	79.00%	
LSTM+SynATT+TarRep	80.63%	71.32
Soft label strategy	80.98%	71.52%
CAN	83.33%	73.23%
ATLS	82.86%	-
AdaRNN	60.42%	-
PhraseRNN	66.20%	62.21%
MemNet	80.95%	-
Tensor DyMemNN	-	58.61%
Holo DyMemNN	-	58.82%
RAM	80.23%	70.80%
Cabse	80.89%	-
FCMN	82.03%	-

Table 3.9 Comparison Study for SEMEVAL 2014 Dataset on Laptop Domain

Method	*Accuracy*	*Macro-F1*
PF-CNN	70.06%	-
PG-CNN	69.12%	-
GCAE	69.14%	-
TNet-LF	76.01%	71.47%
TNet-AS	76.54%	71.75%
IMN	75.36%	72.02%
TD-LSTM	68.10%	68.43%
AE-LSTM	68.90%	62.45%
ATAE-LSTM	68.70%	59.41%
BILSTM-ATT-G	73.12%	69.80%
IAN	72.10%	-
MGAN	75.39%	72.47%
AOA-LSTM	74.50%	-

Table 3.10 Comparison Study for SEMEVAL 2014 Dataset on Laptop Domain

Method	*Accuracy*	*Macro-F1*
AF-LSTM	68.81%	-
PRET+MULT	71.15%	-
Inter-aspect dependencies	72.50%	69.23%
LSTM+SynATT+TarRep	71.94%	-
Soft label strategy	74.56%	71.63%
MemNet	72.37%	52.15%
Tensor DyMemNN	-	55.24%
Holo DyMemNN	-	60.11%
RAM	74.49%	71.35%
Cabse	75.07%	-
FCMN	73.94%	-

Table 3.11 Comparison Study for Twitter Dataset

Method	*Accuracy*	*Macro-F1*
TNet-LF	74.68%	73.36%
TNet-AS	74.97%	73.60%
TD-LSTM	70.80%	69.00%
TC-LSTM	71.50%	69.50%
BILSTM-ATT-G	70.38%	68.37%
MGAN	72.54%	70.81%
AdaRNN	66.30%	65.90%
RAM	69.36%	67.30%
Cabse	71.53%	-

CHAPTER 4

ANALYSIS OF EVOLUTIONARY SOCIAL MEDIA ACTIVITIES: PRE- AND POST-VACCINE EMERGENCY USE

In December 2019, Wuhan, China reported the first case of COVID-19. As the pandemic dramatically spread, the World Health Organization defined this disease as a global pandemic on March 11, 2020. Globally, more than 373 million people have been diagnosed with COVID-19 and 5.66 million have died from this disease by 2022. It continues to have a negative impact on human daily life and the global economic development till now, due to the lack of effective treatment of COVID-19 induced issues and prevention of transmission methods. The COVID-19 has been out of control in most countries due to its highly contagious nature and diverse variants. Research related to COVID-19 is also proceeding at a rapid pace. Ferrag *et al.* [75] provide a discussion of potential solutions to the security and privacy challenges faced in using IoT applications to combat the COVID-19. Ohata [76] uses a transfer learning method to extract features from x-ray images to automatically detect the COVID-19 infection. Developing an effective vaccine to slow down the spread of the outbreak becomes a principal task for researchers and pharmaceutical companies [77]. In 2020, multiple vaccines have received Emergency Use Authorization (EUA) from the U.S. Foods and Drug Administration (FDA), less than a year after the disease was discovered. On December 11, the FDA issued the first EUA for the Pfizer-BioNTech vaccine which benefited individuals of 16 years and older, and vaccination of health care workers started right away. A week later, Moderna COVID-19 vaccine received the second FDA-approved EUA for individuals of 18 years and older. On February 27, 2021, the Johnson & Johnson received vaccine EUA as the third one for those 18 and older ones. All the above vaccine-related information has made people eager to discuss vaccine-related issues. At the same time, massive amounts of information quickly spread on the Internet. Therefore, it is crucial for officials, policy makers and governments to correctly understand the attitudes and concerns of people such that proper policies can be proposed in a more targeted and efficient way and adopted to benefit human combat against this pandemic.

Nowadays, social media platforms, such as Twitter, Facebook and Instagram, provide a faster way which helps people to receive and disseminate information and express personal opinions in real-time. Twitter is a widely used platform that allows the authorized developers to easily obtain tweets using a Twitter API by defining different hashtags and keywords based on various events. In this work, more than one year of COVID-19 vaccine-related twitter data from selected areas are crawled and discussed in detail.

Except analyzing the keywords or special terms in COVID-19 vaccine-related tweets, obtaining human sentiment information from them is even more important. Automatic sentiment analysis is a way to study various aspects such as peoples' emotions, attitudes, and opinions from the target text without the help of humans. The basic task of sentiment analysis is to classify the expressed opinion of a given text into three classes, i.e., positive, neutral, and negative ones. It is not enough to simply observe public attitudes towards vaccines through social networks. More hidden patterns among these tweets need to be explored in detail. Because some tweets discuss topics that are not related to their hashtags, and some tweets do not contain hashtags, topic modeling is one method to solve this problem, and to summarize relevant topics among millions of daily tweets.

This chapter presents a sentiment analysis method to classify a more than one year public discourse about COVID-19 vaccines on Twitter and then reveals the trend of sentiment over time. Also, we present a topic modeling method on tweets to investigate the salient topics for the COVID-19 vaccine in the USA and to discover the trends of different topics before and after vaccination periods. Ultimately based on the findings from the COVID-19 case study's trending topics and sentiment analysis around pre- and post- vaccine tweets, we can improve the readability of confusing messages about vaccines on social media and provide effective results to support government agencies and policymakers.

4.1 Data Collection and Preprocessing

This section focuses on the research of text data. First of all, the COVID-19 vaccine-related twitter data are introduced. Next, noisy data, ambiguous words and redundant information are pre-processed in the data pre-proecessing step.

4.1.1 Dataset Description

This work focus on analyzing the topics and sentiments of the COVID-19 vaccine-related discussion on Twitter, specifically focuses on the period before and after FDA issued EUA for COVID-19 vaccine in United States. The first EUA for COVID-19 vaccine was issued in December 2020. The third one was issued in February 2021. Therefore, tweets posted from March 2020 to May 2021 are selected in this study. The COVID-19 related tweets are collected by using the following 13 keywords: COVID19, CoronavirusPandemic, COVID-19, covid-19, 2019nCov, CoronaOutBreak, coronavirus, WuhanViruscovid19, coronaviruspandemic, 2019ncov, coronaoutbreak, and wuhanvirus.

In this work, we focus on English tweets in the USA. Figure 4.1 shows one tweet related to COVID-19 vaccine on Twitter. Because the twitter user' specific privacy setting policy for geo-reference information, only a limited number of tweets containing geotag, which is not conductive to our analysis of public attitudes and easily causes the sampling bias issue. We assume that users' profile address is the address posting their tweets when their geo-location function is closed. Consequently, we obtain total 118,150,423 COVID-19 related tweets. Vaccine-related tweets are filtered from previous COVID-19 tweets by using the following selected keywords: vaccination, vaccine, immunization, vaccinate, pfizer, biotech, astrazeneca, moderna, J & J, and johnson & johnson. Finally, the total number of 1,647,479 COVID-19 vaccines related tweets that contain at least one of the previous keywords are obtained. The graph of daily vaccine-related count is shown in Figure 4.2. To reduce the noise in the data, we select a 7-day rolling window to smooth the data. We observe that during the initial period of vaccine development, relatively low attention was paid to it. The first peak occurred in November 2020, when FDA issued the first EUA for Pfizer-BioNTech vaccine. Then as vaccination begin and other vaccines were successfully issued, there is an explosion of discussion about vaccine-related events. In 2021, the level of discussion about the vaccines become stabilized.

Figure 4.1 One example tweet related to COVID-19 vaccine.

4.1.2 Data Preprocessing

As a short text message posted online, a tweet has its own formats, structures and properties. Some preprocessing need to be done to clean the data before proceeding to subsequent text analysis. Tweets include some noise such as URL, slang, etc. Such noise may disturb the overall performance of a classification method and reduce the computation speed. For instance, a raw tweet, #obamacare Thanks Obama, Obamacare Would Make The Coronavirus Vaccine Free https://t.co/4awpo5AhxD via @politicususa", has such noise information. Pattern matching is used to check a given pattern sequence of expressions

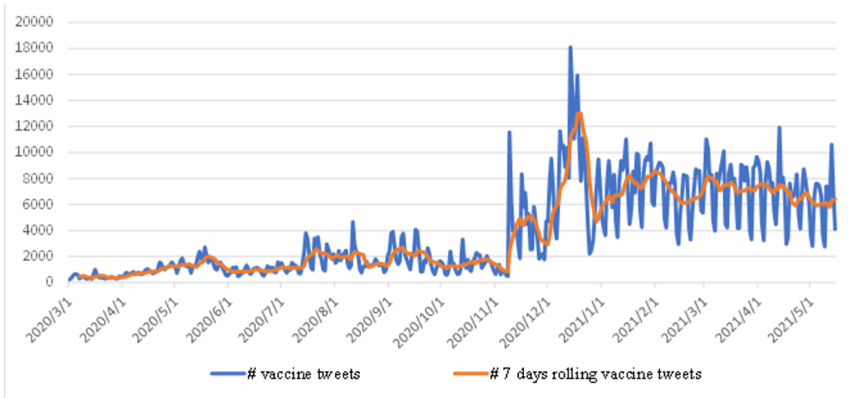

Figure 4.2 Daily tweets count from March 2020 to May 2021.

that match the presence of constituents of some patterns. For example, pattern ”https://” used to catch the URL, and then the information following it is automatically removed until a space is detected. Furthermore, the hashtages, and @ handle references above may not be necessary for topic modeling approach since those terms do not really provide meaningful context for discovering inherent topics from the tweet. Then, lower case all tweets. If the text is in the same case, it is easy for a machine to interpret the words because the lower case and upper case are treated differently by the machine. Therefore, we need to make the text in the same case and the most preferred case is a lower case. Finally, removing stopwords from tweets. Stopwords are the most commonly occurring words in a text which do not provide any valuable information. Stopwords like the, and, it, they, where, etc are some of the stopwords. NLTK library is a common library that is used to remove stopwords. Figure 4.3 shows a summary of our data preprocessing procedure.

4.2 Methodology

In this section, two methods are adopted for in-depth analysis of the COVID-19 vaccine on Tweets. The first one is Valence Aware Dictionary for sEntiment Reasoning (VADER) for sentiment analysis, and the second one is Latent Dirichlet Allocation (LDA) for topic modeling.

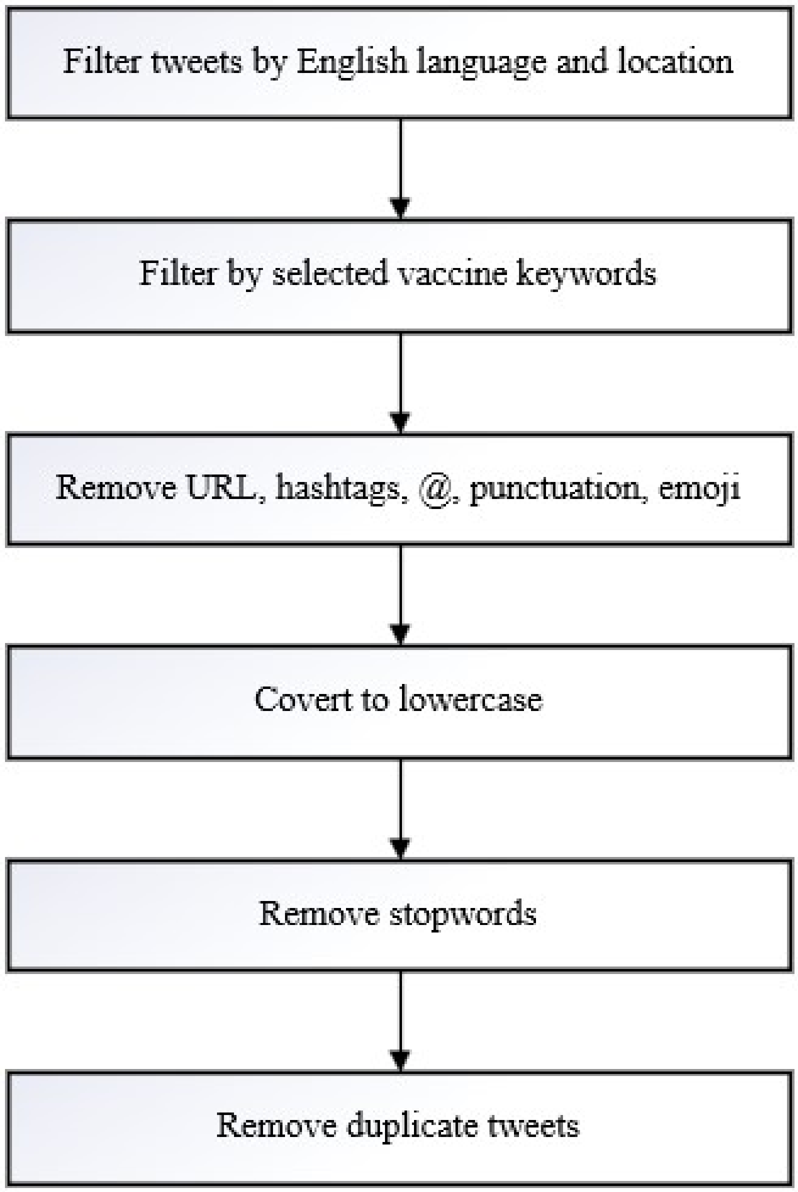

Figure 4.3 Data preprocessing procedure.

4.2.1 Measuring Tweets Sentiment

Sentiment analysis is a way to investigate people's opinion (positive, neutral, and negative) toward text. This work proposes to adopt the Valence Aware Dictionary and sEntiment Reasoner (VADER) [78] to analyze the sentiment of the tweets. VADER is a lexicon and rule-based sentiment analysis tool, which is designed for social media. In this analysis tool, compound score values are used to classify the tweets into three different polarities. A score closer to 1 indicates a more positive opinion view of the text. The most negative opinion is indicated by -1. A tweet's sentiment is positive if its compound score is between 0.05 and 1, neutral between -0.05 and 0.05, and negative between -1 and -0.05. Some example of VADER scoring results are shown in Table 4.1.

Table 4.1 Example of Tweets with Positive/Negative/Neutral Sentiment

Tweets	*Sentiment*
thanks obama obamacare would make the coronavirus vaccine free via	Positive
dated the 24th a vaccine won't stop the new coronavirus the atlantic	Negative
let's say a vaccine is found for the coronavirus are y'all actually gonna get this one	Neutral
what the hell am i exactly supposed to do about the coronavirus video lionelnation n2019 ncov coronavirusupdates hysteria confusion dread weaponized militarized martiallaw quarantine manufactured vaccines panic	Negative
a vaccine won't stop the new coronavirus	Negative
vaccines precious metals and agriculture are looking more intriguing as coronavirus concerns grow globally	Positive

4.2.2 Topic Modeling of Twitter

Topic modeling is a method for the unsupervised classification of documents. Specifically, it is the process of learning, recognizing, and extracting high-level semantic topics across a corpus of unstructured text even when people are unsure what they are looking for. This work adopt to use Latent Dirichlet Allocation (LDA) [79], which is a generative probabilistic topic model, to identify the salient topic from tweets. The data-driven and computational natural of LDA makes it attractive for this study, as it can be used to quickly and efficiently derive the topic structure of a large number of text files.

Bayes' theory Bayes' theory is an important mathematical concept involved in the LDA model. It is mainly used to infer the probability of an event probability (in case another event is known to be associated with this event). If expressed in terms of two conditional probabilities, for instance $P(A|B)P(B|A)$, according to the multiplication law: $P(A \bigcap B) = P(A) * P(B|A) = P(B) * P(A|B)$, which can be derived immediately, then using Bayes' formula expressed as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{4.1}$$

According to Bayes' rule, $P(A)$ represents the prior or marginal probability of event A, because it does not take into consideration of event B. $P(A|B)$ is the conditional probability of A occurring after B is known, and is also the posterior probability of A. The meaning of $P(B|A)$ is similar to the meaning of $P(A|B)$. $P(B)$ is the prior or marginal probability of B. It is also used as a normalized constant. Thus, Bayes' law can be expressed as: $posterior\ probability = \frac{likelihood * prior\ probability}{normalized\ constant}$.

Dirichlet distribution The Dirichlet distribution $Dir(\alpha)$ is a family of continuous multivariable probability distributions parameterized by a vector α of positive reals. Dirichlet distributions are commonly used as prior distributions in Bayesian statistics.

The probability density function of Dirichlet distribution as following:

$$Dir(\theta|\alpha) = \frac{1}{Beta(\alpha)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} \tag{4.2}$$

where,

$$Beta(\alpha) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{K} \alpha_i)} \tag{4.3}$$

The Dirichlet distribution is parameterized by the vector $\alpha = (\alpha_1, \alpha_2, ..., \alpha_k)$, which has the same number of elements K as our multinomial parameter θ. K is the number of variables.

LDA model The basic idea of LDA is that each topic is characterized by a distribution over words, and each document is represented by a distribution over latent topics. Figure 4.4 shows a graphical model for LDA. Table 4.2 shows the meaning of the notations in LDA.

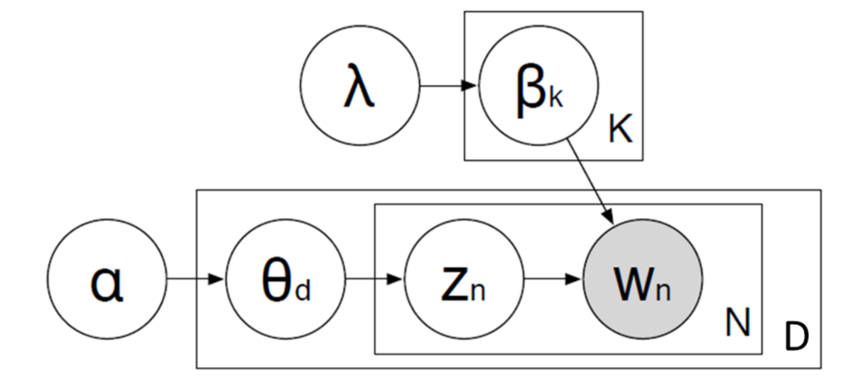

Figure 4.4 Graphical model for LDA.

The generative process is presented as follows:

1. For each topic k, sample a distribution over words: $\beta_k \sim Dir(\lambda)$; and

2. For each document d with N words:

a) Sample a distribution over topics: $\theta_n \sim Dir(\alpha)$

b) For each word $w_{d,n}$: Sample a topic $z_{d,n} \sim Multi(\theta_d)$ and sample a word $w_{d,n} \sim Multi(\phi_{z_{d,n}})$

Table 4.2 Meaning of the Notations

Symbol	*Description*
K	total number of topics
D	total number of documents
N	total number of words in a document
α, λ	Dirichlet parameters
θ_d	per-document topic proportions
Z_n	per-word tropic assignment
W_n	observed word
β_k	topic-word distribution for topic k

The joint probability distribution among all variables can be calculated as:

$$P(\vec{w_d}, \vec{z_d}, \vec{\theta_d}, \beta|\vec{\alpha}, \vec{\lambda}) = P(\theta|\vec{\alpha}) \prod_{n=1}^{N_d} P(w_{d,n}|\vec{\beta}_{z_{d,n}}) P(z_{d,n}|\vec{\theta}_{d,n}) P(\vec{\theta}_{d,n}|\vec{\alpha}) \tag{4.4}$$

For single document $\vec{w_d}$, the probability is:

$$P(\vec{w_d}|\vec{\alpha}, \vec{\lambda}) = \int \int P(\vec{\theta_d}|\vec{\alpha}) P(\beta|\vec{\lambda}) \prod_{n=1}^{N_d} P(w_{d,n}|\vec{\theta_d}, \beta) d\beta d\vec{\theta} \tag{4.5}$$

The probability of entire document is:

$$P(W|\vec{\alpha}, \vec{\lambda}) = \prod_{d=1}^{D} P(\vec{w_d}|\vec{\alpha}, \vec{\lambda}) \tag{4.6}$$

LDA is an unsupervised method, meaning that the number of topics in the corpus is unknown. A low number of topics produce a general classification result of topics, which not conductive to deeper exploration of the underlying topic. Many topics indicate that the classification results are too fine to be summarized, thus requiring further topic aggregation. This work proposes to apply the Python package Gensim to evaluate the optimal number of topics in the LDA model. We test the number of topics range from 2 through 30. For each topic number, we calculated the coherence score. The coherence score helps to distinguish between human understandable topics and artifacts of statistical inference, which is calculated as follows:

$$Coherence = \sum_{i<j} score(w_i, w_j) \tag{4.7}$$

The coherence selects top n frequently occurring words in each topic, then aggregates all the pairwise scores of the top n words $w_i, ..., w_n$ of the topic. Figure 4.5 displays the coherence score of all tweets for the number of topics across two validation sets, and a fixed $\alpha = 0.01$ and $\beta = 0.1$. We ended up choosing 13 topics for the final model, as the topic number that was equal to 13 yielded the highest coherence score.

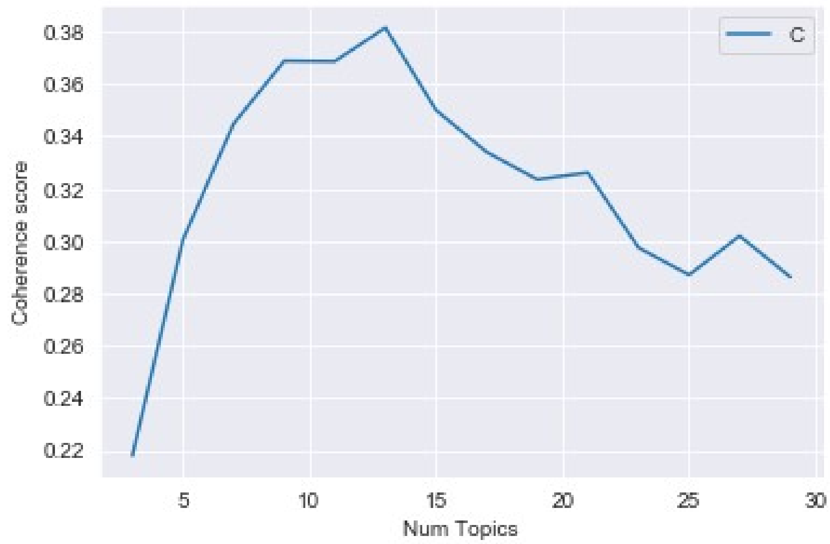

Figure 4.5 Coherence score corresponding to the different number of topics.

Gibbs sampling Gibbs sampling is a parameter estimation method that is mostly applied in LDA model for parameter estimation. It is an efficient way of reducing a multi-dimentional problem to a lower-dimensional problem.

In LDA, Gibbs sampling assumes that the currently word assignment of a topic is influenced by the words assignment of other topics, which means that we are trying to find conditional probability distribution of a single word's topic assignment conditioned on the rest of the topic assignments. The conditional probability for a single word w in document d that belongs to topic k is calculated as follows:

$$P(z_{d,n} = k|\vec{z}_{-d,n}) = \frac{n_{d,k} + \alpha_k}{\sum_i^k n_{d,i} + \alpha_i} \frac{v_{k,w_{d,n}} + \lambda_{w_{d,n}}}{\sum_i v_{k,i} + \lambda_i} \tag{4.8}$$

where $n_{d,k}$ means the number of times document d use topic k, $v_{k,w}$ means the number of times topic k uses the given word, α_k and λ_w are the Dirichlet parameter for document to topic distribution and topic to word distribution, respectively.

4.3 Experimental Results

Our experiments use an Intel Core i7-8700 CPU @ 3.20GHz, NVIDIA GeForce GTX1080 GPU, and Windows 10 64bit. We use Python 3.0, NLTK 3.3, Spacy 2.3, and Gensim 4.1 to realize the experiments.

Figure 4.6 shows the number of vaccine tweets per day for positive, neutral, and negative, respectively. In the few discussions about vaccines in March and April 2020, only a small percentage of people have positive attitudes towards vaccine development. Between April and November 2020, although the number of positive attitudes exceeded the number of negative ones, there is no clear distinction between them. Until the first vaccine is approved in the beginning of November 2020. The positive attitudes on the Internet substantially increased.

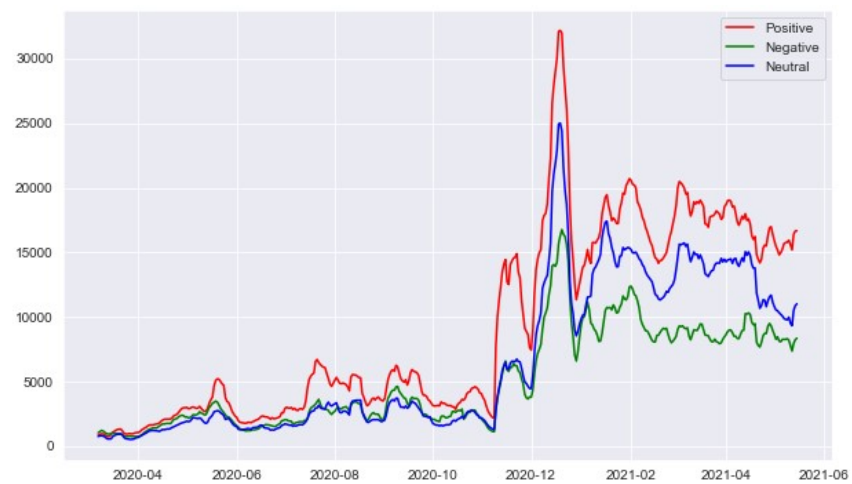

Figure 4.6 Sentiment information for daily tweets from March 2020 to May 2021.

Tables 4.3 - 4.4 presents the topic modeling results. It shows the most 15 probable terms for the given 13 topics. Table 4.5 shows the number of tweets percentage in each topic. Topic 2 and Topic 9 share 12.97% and 12.57% of all tweets respectively, which means these two topics contain the highest attention. According to the high-frequency terms in Topic 2, it focuses on the instruction of getting vaccine. Table 4.6 summarizes the top five frequency topics, and provides a tweet example for each topic.

Table 4.3 Top 15 Frequencies Term per Topic

Topic	*Terms per Topic*
Topic 1	country, world, test, life, india, fact, leader, effect, proof, way, travel, man, choice, fear, need
Topic 2	appointment, county, week, site, clinic, resident, today, age, pm, april, dose, state, march, department, center
Topic 3	question, information, help, sign, expert, location, answer, info, detail, check, passport, link, resource, visit, today
Topic 4	mask, child, cdc, thing, employee, kid, school, family, friend, parent, vaccineswork, business, schedule, wear, mandate
Topic 5	case, death, risk, rate, number, population, infection, disease, line, data, cdc, herd_immunity, day, control, record
Topic 6	year, month, flu, end, time, science, hope, pandemic, work, scientist, symptom, reason, week, trust, evidence
Topic 7	biden, trump, president, variant, administration, york, government, china, america, hesitancy, joe, effort, white_house, plan, day
Topic 8	shot, day, woman, group, immunity, arm, today, nd, yesterday, hour, person, st, word, love, reaction
Topic 9	pfizer, johnson, moderna, dose, trial, fda, use, study, emergency, jampj, mrna, astrazeneca, uk, week, datum

Table 4.4 Top 15 Frequencies Term per Topic

Topic	*Terms per Topic*
Topic 10	community, worker, care, student, patient, staff, healthcare, member, hospital, school, access, teacher, provider, team, college
Topic 11	news, dr, thank, fauci, card, research, video, share, ohio, director, morning, post, watch, good, experience
Topic 12	state, distribution, plan, medicine, official, pharmacy, governor, texas, california, doctor_policy, guidance, pause, supply, system, guideline
Topic 13	americans, rollout, safety, article, company, development, side_effect, treatment, race, drug, research, progress, survey, market, jab

Figure 4.7 shows how the different topics flow in the timeframe of March 2020 to May 2021. The y-axis shows the percentage of 13 topics in the same day. By changing the percentage over time, we can observe the flow of these topics and gain insight into what topics people are paying more attention to at different stages. For example, in Topic 2, which has the largest overall weight. It can be observed that the Topic 2 has experienced a dramatic increase in the amount of attention after November 2020, when FDA issued the first EUA for Pfizer-BioNtech vaccine. As the number of vaccinations increases, the amount of discussion on this topic begins to decrease. Topic 9 is most discussed in the pre-vaccine emergency use phase, which is also the development phase of vaccine. As vaccines were approved one after another, people no longer had interest in this topic. On the contrary, people started to pay attention to the topic related to vaccination. Topic 10 and Topic 2 have similar trends, as both topics discuss content related to vaccine registration. Topic 7 related to the U.S. President and government, which received the most discussion during the election period. After the election day ends, the percentage of this topic begins to decrease.

Table 4.5 Tweets Percentage per Topic

Topic	*# Tweets*	*Percentage*
Topic 1	122,259	9.29%
Topic 2	170,680	12.97%
Topic 3	74,955	5.70%
Topic 4	74,955	5.70%
Topic 5	89,975	6.84%
Topic 6	92,459	7.03%
Topic 7	103,588	7.87%
Topic 8	58,710	4.46%
Topic 9	165,489	12.57%
Topic 10	108660	8.26%
Topic 11	72,713	5.52%
Topic 12	83,578	6.35%
Topic 13	91,805	6.98%

Table 4.6 Top five Topics

Topic	*Themes*	*Example of tweets*
Topic 2	Instruction of getting vaccine	the covid19 vaccine call center will be closed from friday april 2nd until sunday april 4th it will reopen on monday april 5th at 830 am for updates please visit
Topic 9	Clinical trails	johnson amp johnson files for emergency authorization for coronavirus vaccine
Topic 1	Worldwide opining of getting vaccine	it is a new world order hoax designed and released to panic the entire world and issue in the new vaccine that will be designed to depopulate kill off a lot of the worlds people
Topic 10	vaccine rollout	full endorse schools as vaccination hubs teachers school personnel families parentsone way to ensure that teachers and school personnel have clear access to the vaccine
Topic 7	American president	joe biden is trying to take credit for the vaccine of the coronavirus excuse me donald trump is still our president

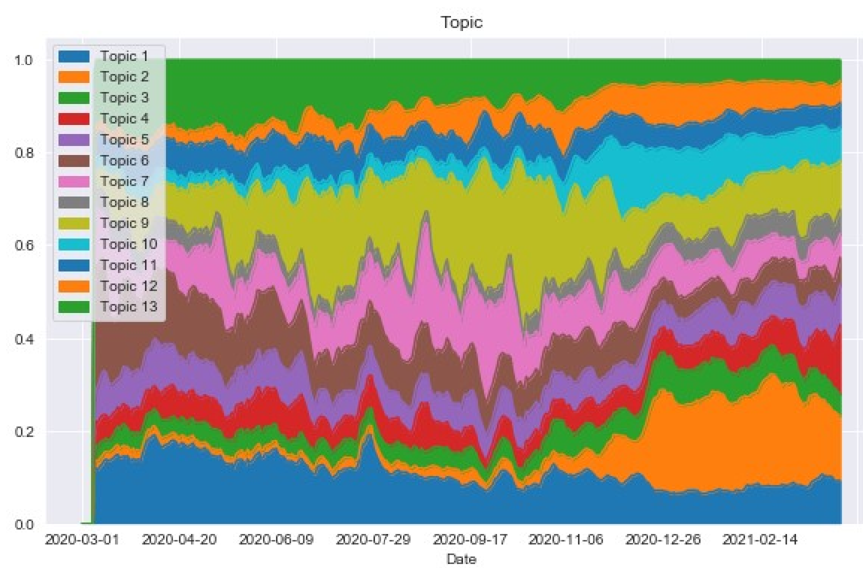

Figure 4.7 Topic flow trend between the March 2020 and May 2021.

Figures 4.8 - 4.13 show the trend of sentiment from topic 1 to topic 13. We can find that people's attitude toward positive under most of topic. Especially more brand of vaccines are issued, the positive attitude has a rapid growth. However, there are some topics where the negative attitude is heavily weighted.

Figure 4.8 Sentiment trend for topic 1 and topic 2.

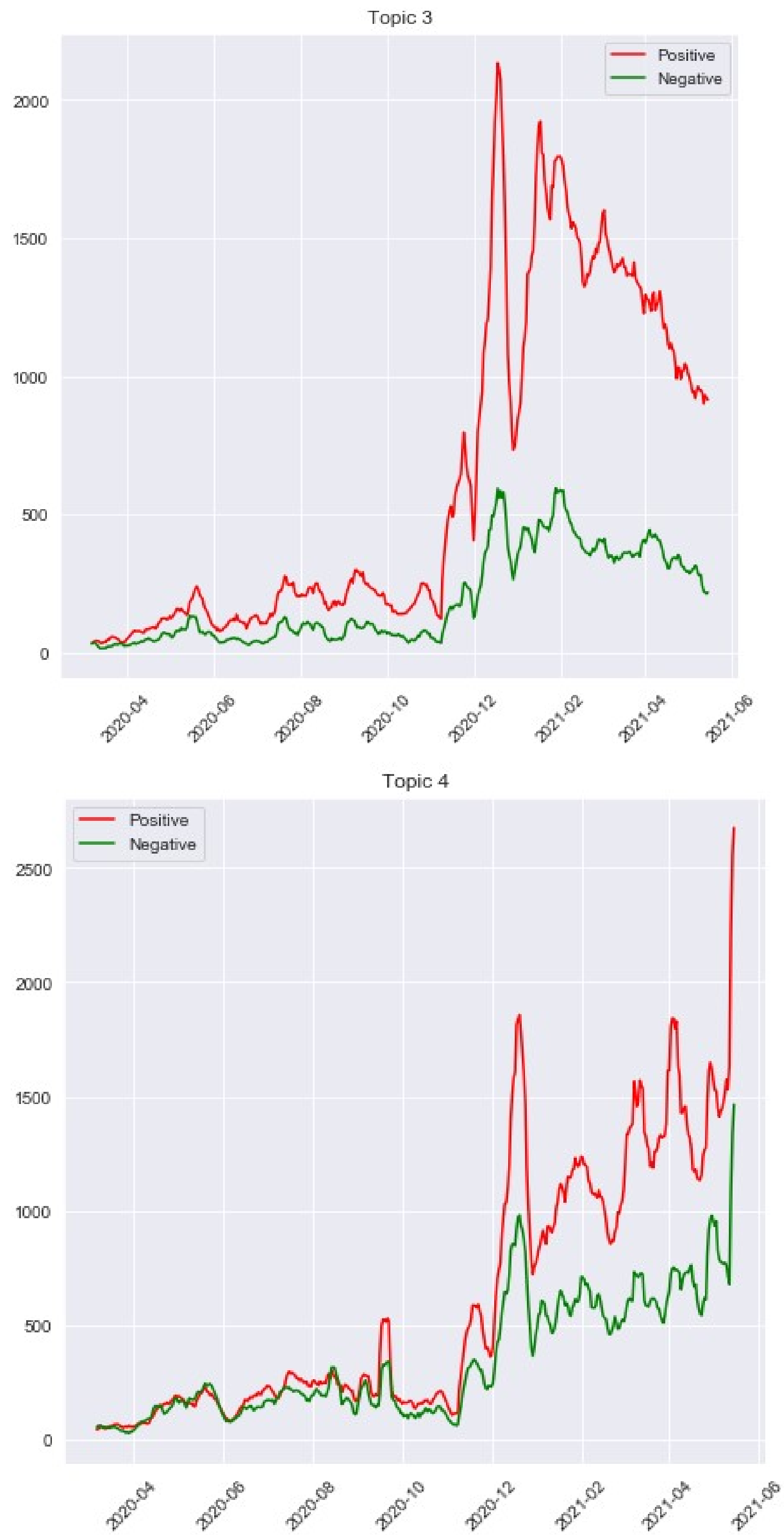

Figure 4.9 Sentiment trend for topic 3 and topic 4.

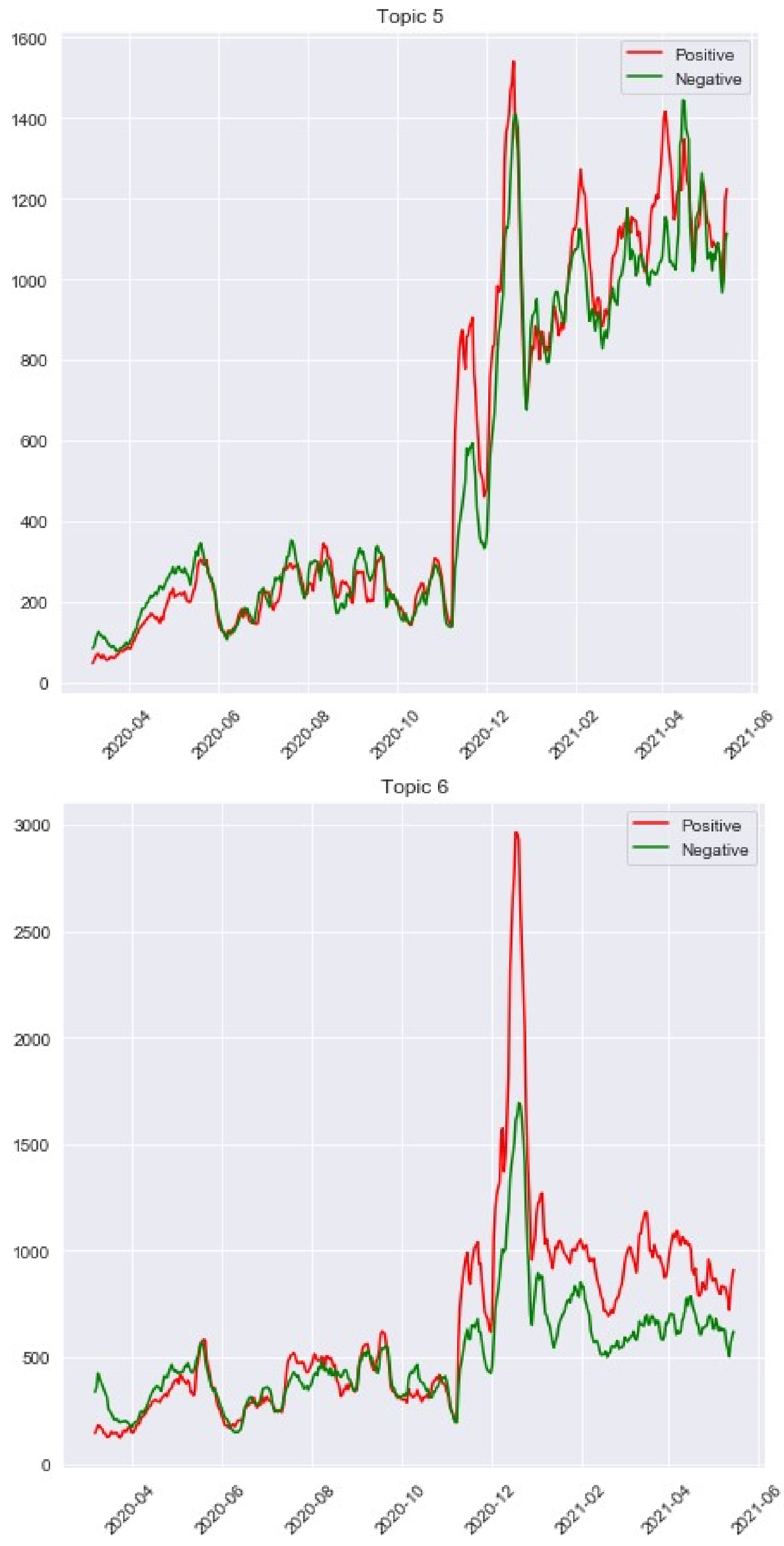

Figure 4.10 Sentiment trend for topic 5 and topic 6.

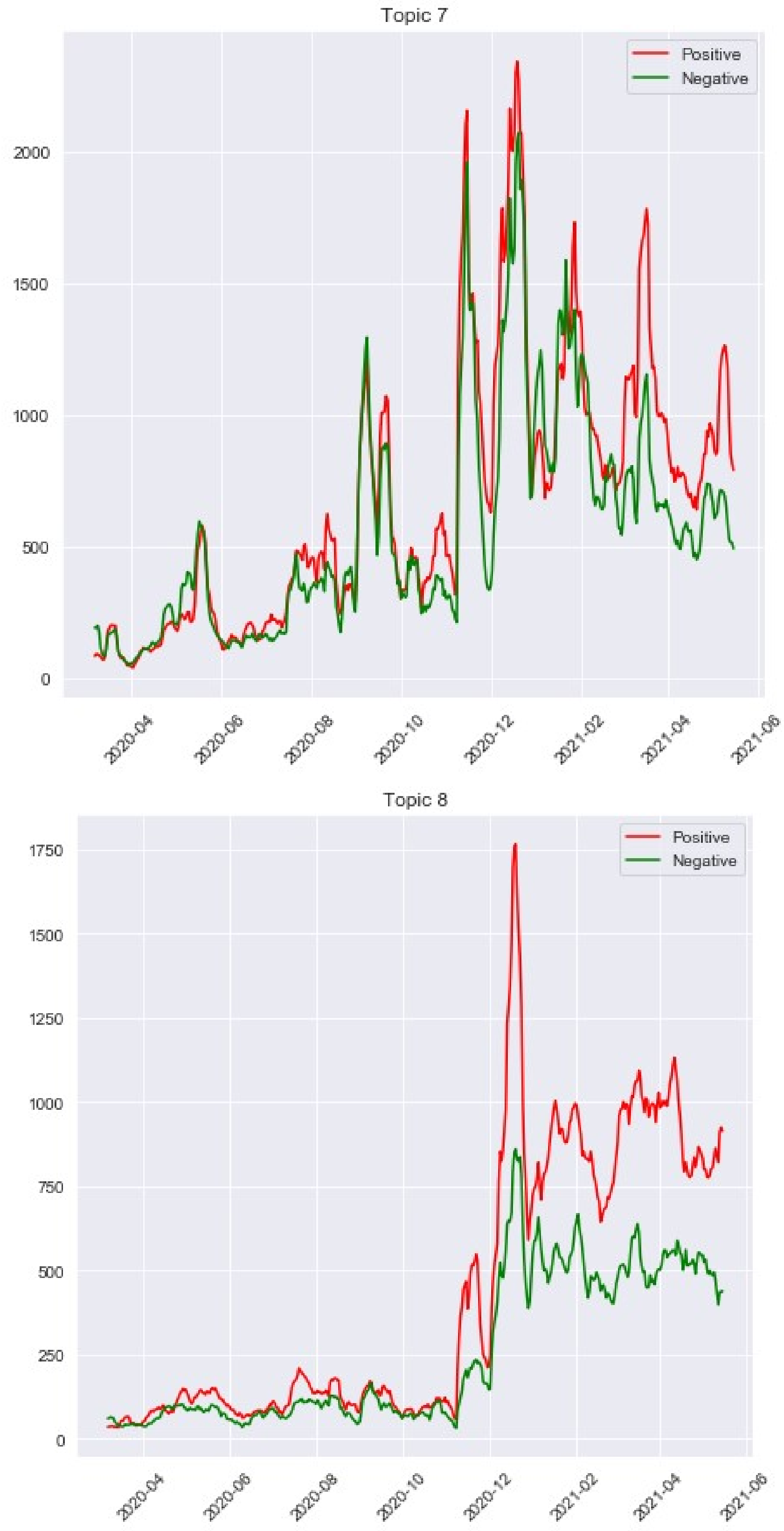

Figure 4.11 Sentiment trend for topic 7 and topic 8.

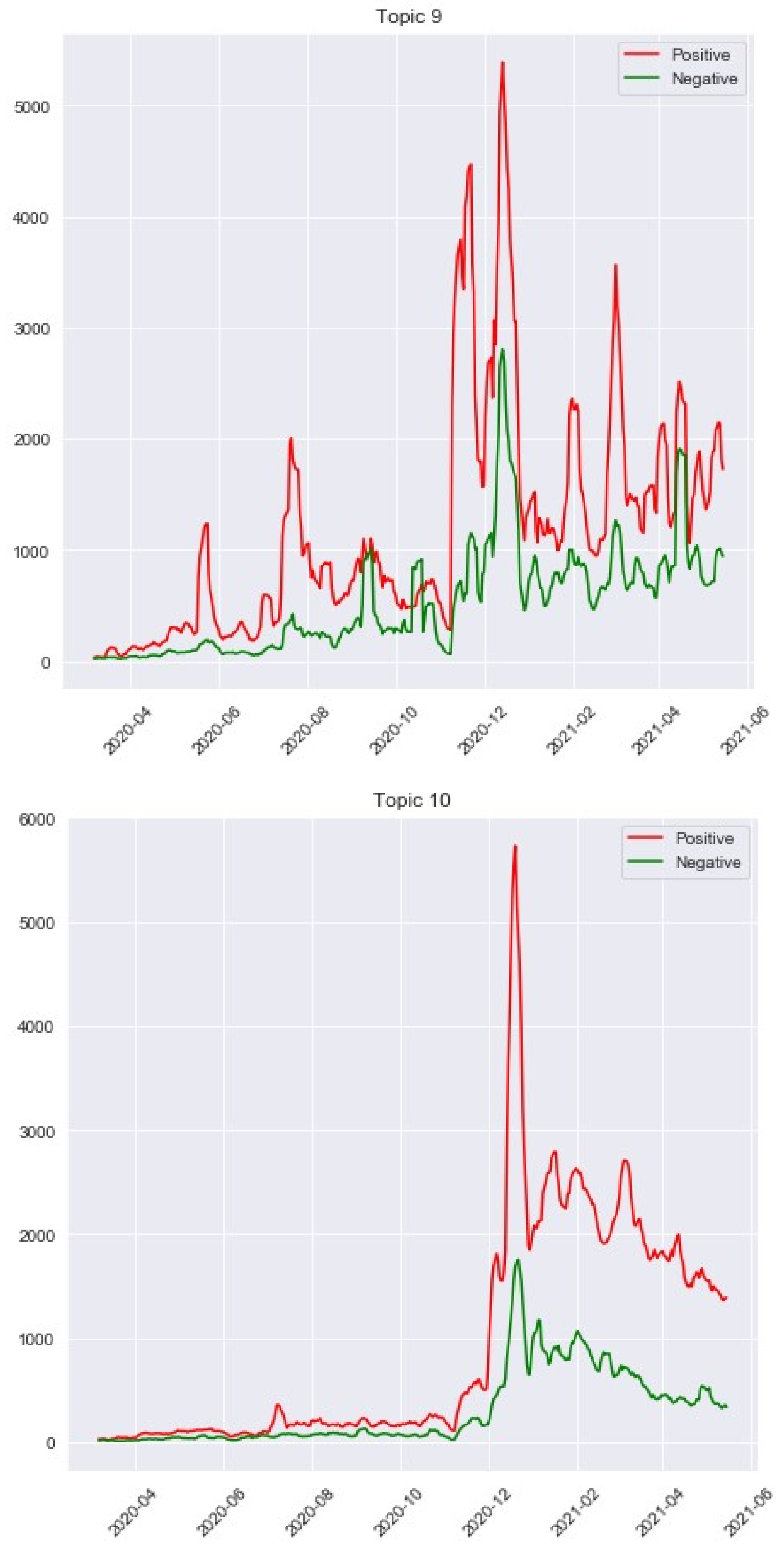

Figure 4.12 Sentiment trend for topic 9 and topic 10.

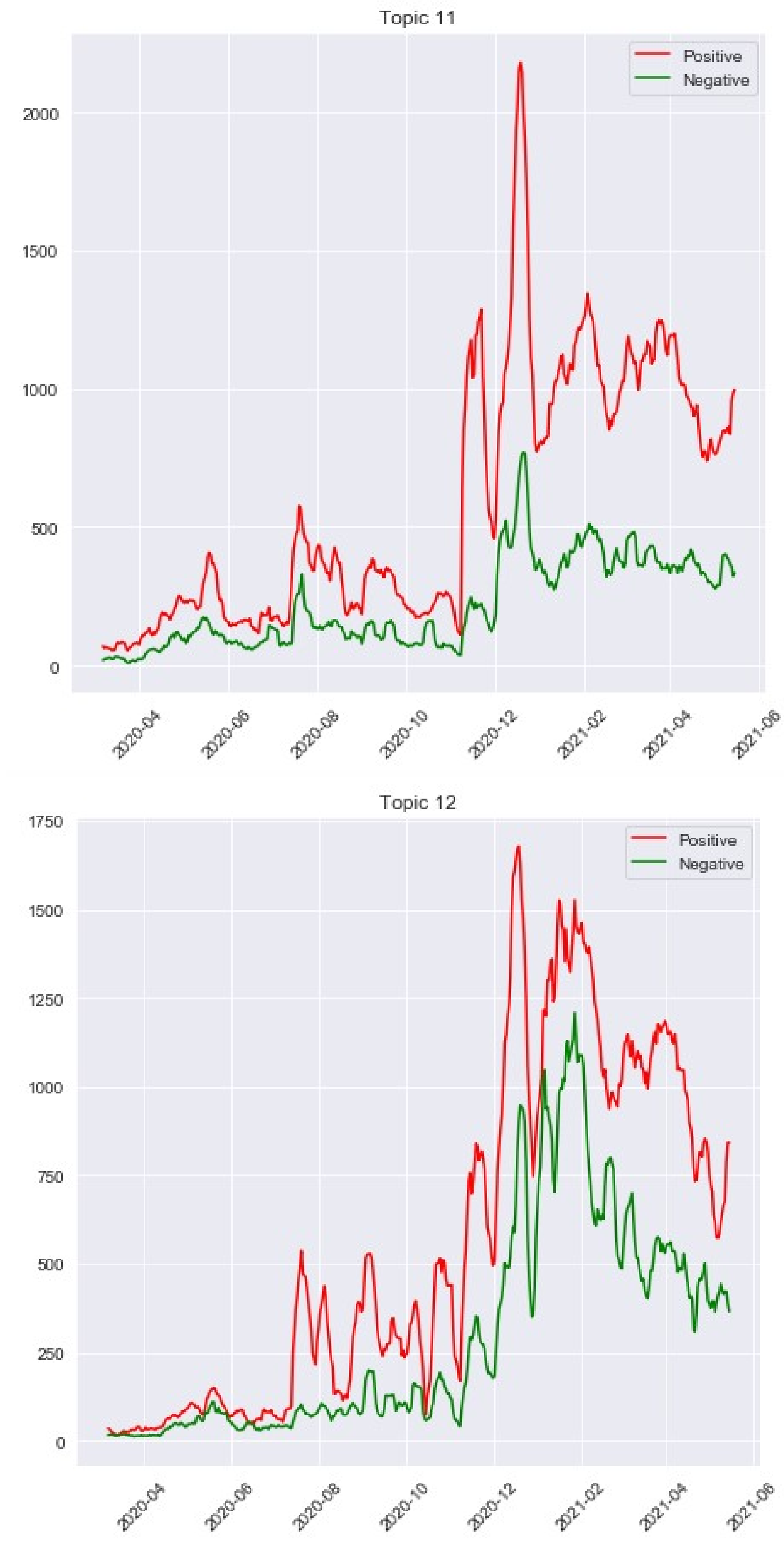

Figure 4.13 Sentiment trend for topic 11 and topic 12.

Figure 4.14 Sentiment trend for topic 13.

CHAPTER 5

IMPROVING SHORT TEXT TOPIC MODEL WITH WORD EMBEDDINGS

In the previous chapter, we adopt LDA model to extract topics information from short texts. However, short texts suffer from severe data sparsity problem due to the limited length of short texts. This kind of data sparsity problem will cause LDA model fail to extract coherent topics from short texts.

Several strategies have been proposed to deal with the data sparsity problem in short texts. One strategy is to aggregate a subset of short texts to form a longer pseudo-document by using auxiliary information. Although additional metadata information may not be available always. The second strategy is to restrict the document-topic distribution, such as Dirichlet Multinomial Mixture (DMM) [17], assuming each short text sampled from a single topic. This assumption is too strong in reality due to the nature of short text. The third strategy mainly utilize word co-occurrence information, such as Biterm Topic Model (BTM) [18] uses co-occurred word pairs mined from corpus for topic inference. However, all the above improved strategies only utilize the internal information of corpus while ignoring the external knowledge. Word embeddings are widely studied in recent years, aiming at incorporating auxiliary information to promote the semantically related words under the same topic [21].

In this work, we utilize both semantic similarity information from word embedding and word co-occurrence information in the corpus to mine topic inference. Also, there are some noise are contained in sparse social media posts. Churchill and Singh [80] propose a topic-noise models (TND) which define a document as a mixture of topics and noise, and then combining TND with LDA to create high-coherence, high-diversity, low-noise topics.

Motived by the [21], we proposed a novel topic model, named TN-BTMF (Topic Noise based Biterm Topic Model with FastText embeddings), which is designed to alleviate the sparsity problem by exploiting both semantic similarity information and word co-occurrence during the sampling process. And then assuming a document as a mixture of topics and noise to alleviate the noise problem in short texts. Finally, Generalized $P\acute{o}lya$ Urn model (GPU) [21] is employed for sampling.

5.1 Biterm Topic Model

The key idea of Biterm Topic Model (BTM) [18] is to learn topics over short texts based on the aggregated biterms in the whole corpus to tackle the sparsity problem in single document. Figure 5.1 - 5.2 shows the graphical representation of LDA and BTM. The major different between LDA and BTM is BTM consider that the whole corpus as a mixture of topics.

The generative process of the corpus in BTM can be described as follows:

1. For each topic z

   draw a topic-specific word distribution $\beta_z \sim Dir(\lambda)$

2. Draw a topic distribution $\theta \sim Dir(\alpha)$ for the whole corpus

3. For each biterm b in the biterm set B

   draw a topic assignment $z \sim Multi(\theta)$

   draw two words $w_1, w_2 \sim Multi(\beta_z)$

The joint probability of a biterm $b = (w_i, wj)$ can be written as:

$$P(b) = \sum_z P(z)P(w_i|z)P(w_j|z) = \sum_z \theta_z \beta_{i|z} \beta_{i|z} \tag{5.1}$$

Thus the likelihood of the whole corpus is:

$$P(B) = \prod_{(i,j)} \sum_z \theta_z \beta_{i|z} \beta_{i|z} \tag{5.2}$$

5.1.1 Inferring Topics in Document

Unlike LDA generation process, BTM does not model the document generation process. Therefore, the topic proportions cannot directly obtained during the topic learning process. To infer the topics in a document, BTM assumes that the topic proportions of a document

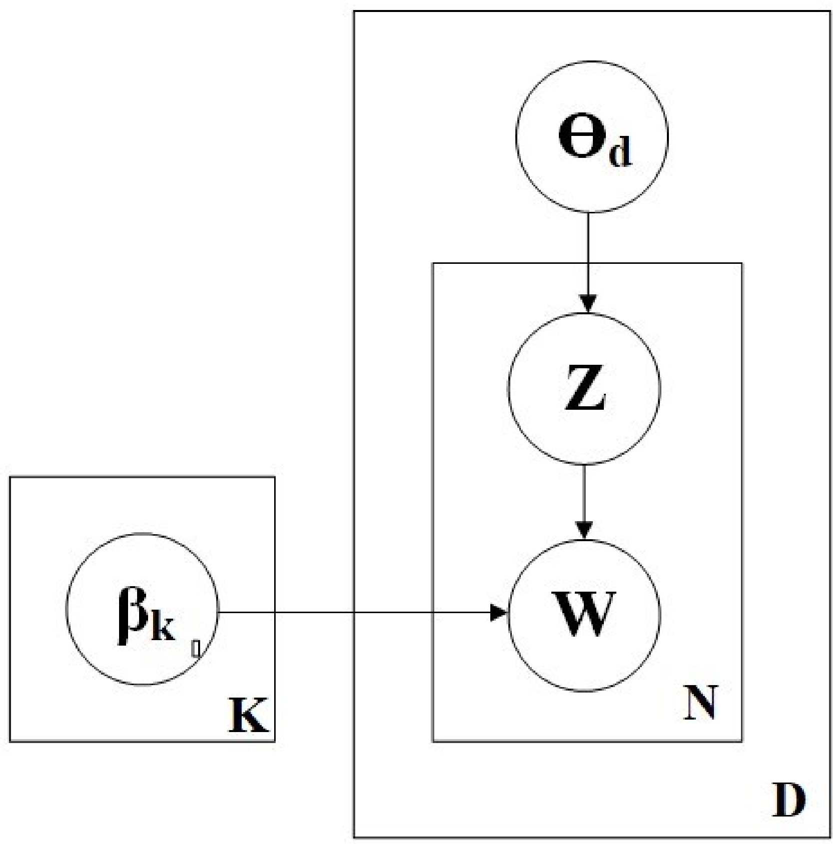

Figure 5.1 Graphical representation of LDA.

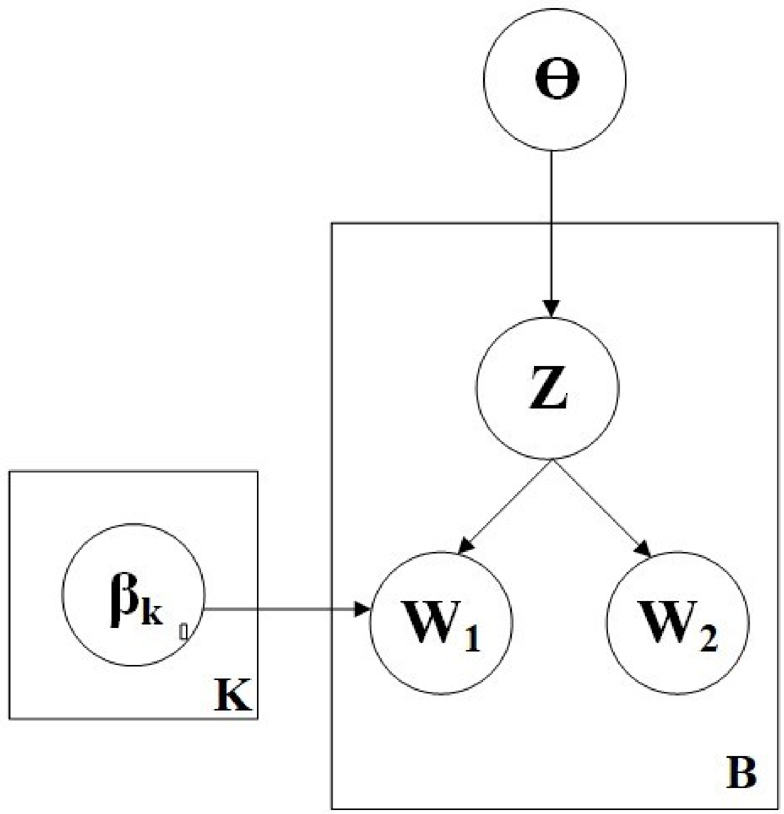

Figure 5.2 Graphical representation of BTM.

equals to the expectation of the topic proportions of biterms generated from the document:

$$P(z|d) = \sum_b P(z|b)P(b|d) = \frac{P(z)P(w_i|z)P(w_j|z)}{\sum_z P(z)P(w_i|z)P(w_j|z)} \tag{5.3}$$

where $P(z) = \theta_z$, and $P(w_i|z) = \beta_{i|z}$. Then $P(b|d)$ is estimated as:

$$P(b|d) = \frac{n_d(b)}{\sum_b n_d(b)} \tag{5.4}$$

where $n_d(b)$ is the frequency of the biterm b in the document d.

5.1.2 Inference by Gibbs Sampling

In BTM, we need to sample all the three types of latent variables z, β, and θ. According to the technique of collapsed Gibbs sampling, β, and θ can be integrated out due to the conjugate priors λ and α. Consequently, we only have to sample the topic assignment for each biterm from its conditoinal distribution given the remaining variables.

$$P(z|z_{-b}, B, \alpha, \lambda) \propto (n_z + \alpha)\frac{(n_{w_{i|z}} + \lambda)}{(\sum_w n_{w|z} + M\lambda)^2} \tag{5.5}$$

where n_z is the number of times of the biterm b assigned to the to the topic z, the two words w_i and w_j in it will be assigned to the topic simultaneously.

The posterior probabilities of topic distribution and topic-word distribution can be calculated by point estimation as follows:

$$\beta_{w|z} = \frac{n_{w|z} + \lambda}{\sum_w n_{w|z} + M\lambda} \tag{5.6}$$

$$\theta_z = \frac{n_z + \alpha}{|B| + K\alpha} \tag{5.7}$$

where B is the total number of biterms.

5.2 Noise Biterm Detection Method based on Words Co-occurrence and Semantic Relationships

In this section, a scoring method based on words co-occurrence and semantic similarity is proposed to detect noise biterms. When a biterm contians a high frequency, but the two words are not show semantically relatedness, such noise biterm will degrade the performance of topic inference.

5.2.1 TextRank for Word Co-occurrence

The TextRank algorithm is a variant of PageRank, which represents the text as a graph, and measures the importance of words according to all words in the text. Formally, let $G = (V, E)$ be a graph, where V is denoted as words in short texts, and E is a set of edges based on the co-occurrence of words, the weight of edge is calculated by the number of co-occurrence between two words in the same tweet. TextRank calculates the score of nodes in the graph is as follows:

$$R(V_i) = (1 - d) + d \cdot \sum_{j:V_j \to V_i} \frac{w_{j,i}}{\sum_{k:V_j} w_{j,k}} R(V_j) \tag{5.8}$$

where $w_{j,i}$ is the weight of edge from node V_j to the current node V_i and $\sum_{k:V_j} w_{j,k}$ is the summation of all edge weight in the previous nodes V_j. d is the damping factor which denotes the probability of randomly selecting one node in the graph, and the value is usually set to 0.85.

$P(V_j, V_i)$ represents the probability of a biterm $b(V_j, V_i)$, which is calculated as:

$$P(V_j, V_i) = \frac{w_{j,i}}{\sum_{k:V_j} w_{j,k}} \tag{5.9}$$

5.2.2 Semantic Relationships

The weight of the edges can affect the final biterm probability. Different methods for edge weight calculation can get different word graphs. In this work, we use semantic relationships to establish weighed edges between any semantically related word.

Many deep learning models in NLP are based on word embedding as input features. In this work, we exploit FastText to learn learn word embeddings. FastText can efficiently can efficiently train on large-scale datasets, and the word embeddings learn by this tool are able to capture the similarity between words. The semantic similarity between words as:

$$sim(w_i, w_j) = \frac{v_i \cdot v_j}{\|v_i\| \cdot \|v_j\|} \tag{5.10}$$

where, v_i and v_j are word vectors corresponding to word w_i and w_j respectively. of the notations in LDA. Figure 5.3 shows the semantic similarity result for word ”vaccine” ”obama” and ”mask” based on FastText pre-trained word embedding.

5.2.3 Noise Biterm Detection Method

We use the semantic similarity between words in biterms to weight the edges of these biterms. The edge's weight can be calculated as:

$$weight(w_i, w_j) = freq(w_i, w_j) \cdot semantic(w_i, w_j) \tag{5.11}$$

where $freq(w_i, w_j)$ is the frequency of the biterm occure in same tweet. The calculation of $P(V_j, V_i)$ implies that if a biterm has higher value, it is less likely as a noise biterm.

5.2.4 Topic-Noise Model

Topic Noise Discriminator (TND) [80] is a topic-noise model that estimates both the topic and noise distributions simultaneously. Let D represent a dataset consisting of M posts, where $D = d_0, d_1, ..., d_{M-1}$. A document d is collection of N words, where $d = w_0, w_1, ..., w_{N-1}$. A topic t consists of a set of l words, $t = w_0, w_1, ..., w_{l-1}$ A topic set T consists of k topics, where $T = t_0, t_1, t_{k-1}$. A noise set H consists of a set of p words, $H = w =_0, w_1, ..., w_p$, where the words in H represent noise.

The decision of whether a biterm is a noise term is determined using the Beta distribution. The Beta distribution λ is the special case of the Dirichlet where $k = 2$, and x controls whether the biterm is draw from Z or H. This distribution is controlled

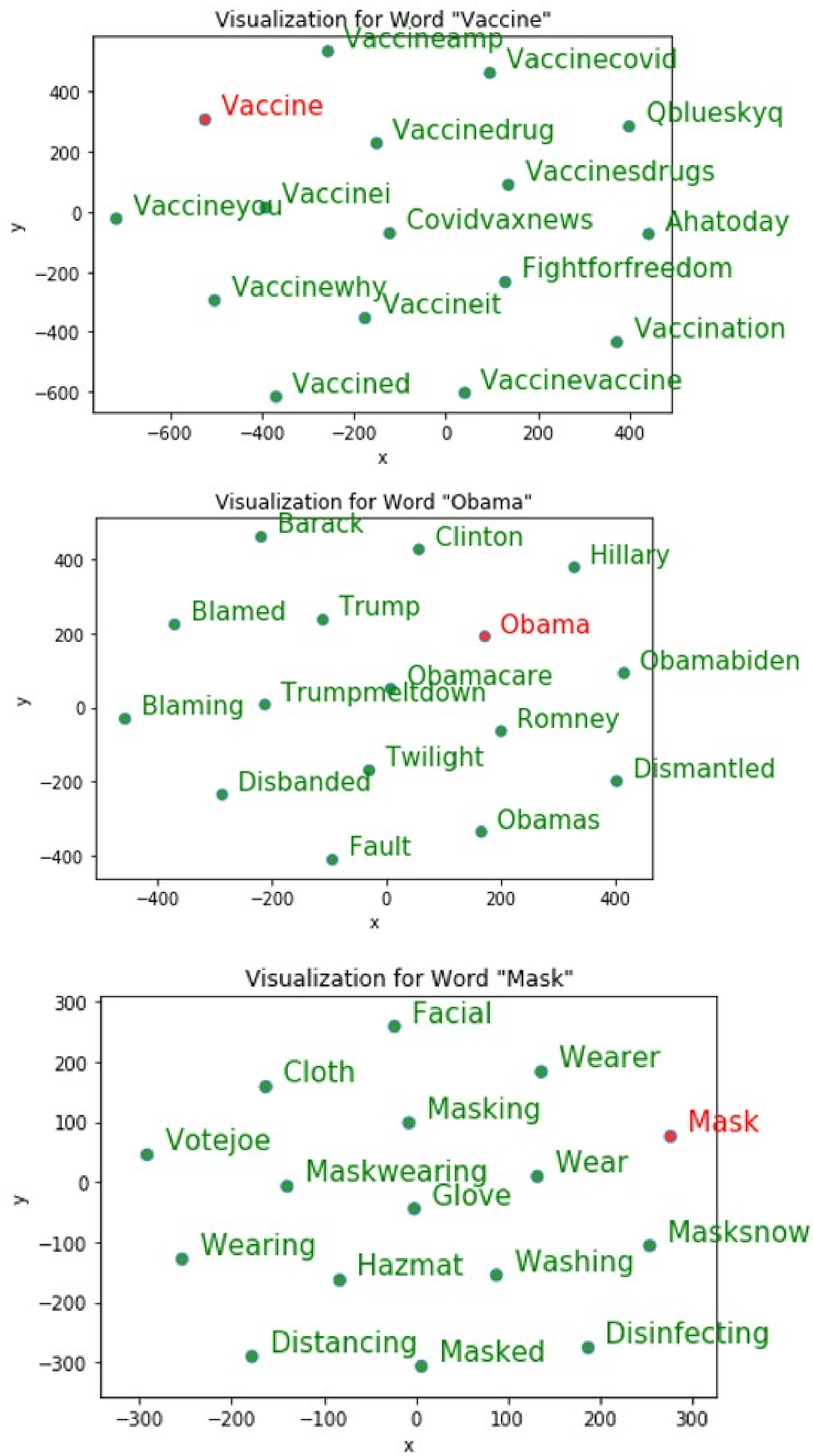

Figure 5.3 Graphical semantic similarity result for ”vaccine” ”obama” and ”Mask”.

on the β_1 parameter. The initial value of β_1 need to set higher that allows us to skew the distribution to favored topic if the expectation of noise is less than topics.

5.3 GPU Embedding Sampling

First, we extract the biterm set as a whole text, and then collect the similar words by using cosine similarity. Word representation can be done with auxiliary corpus and FastText model. Lastly, Gibbs sampling with GPU model is utilized to do model inference. Figure 5.4 presents the graphical model for TN-BTMF.

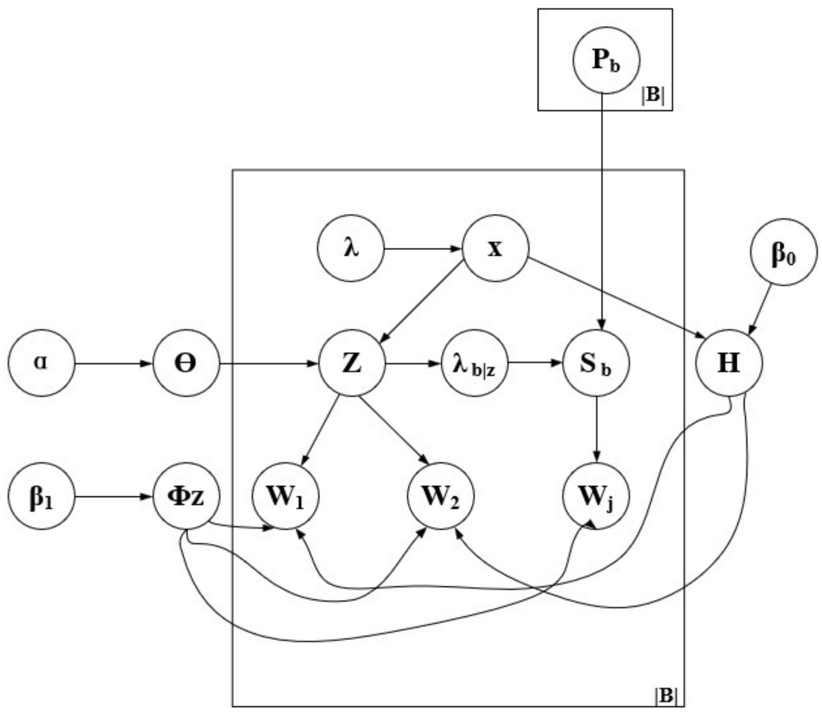

Figure 5.4 Graphical TND for BTM.

The generative process for TN-BTMF is as follows:

1. For each topic z

2. Draw a topic-specific word distribution $\phi_z \sim Dir(\beta)$

3. Draw a topic distribution $\theta \sim Dir(\alpha)$ for the whole biterm collection

4. For each biterm b_i in the biterm set B

5. Draw the topic of $b_i, z \sim Multi(\theta)$

6. Draw a biterm b_i from the either z_i or the noise distribution H, according to the Bernoulli distribution, conditioned on α

7. If drawing from z_i, draw two words w_1, w_2 based on the probability of b_i given the topic z_i and conditioned on β_0

8. If drawing from H, draw two words w_1, w_2 based on the probability of b_i given the topic H and conditioned on β_1

The probability of drawing a topic for a biterm b depends on two latent variable: z_b and I_b is calculated as follows:

$$p(I_b = k, z = k | I^{-b}, z^{-b}, B, \alpha, \beta_1, p_b) = (1-p_b) \frac{m_k^{-b} + \alpha}{\sum_{k=1}^{K} m_k^{-b} + \alpha} \frac{(n_{w_{1|k}}^{-b} + \beta)(n_{w_{2|k}}^{-b} + \beta)}{(n_k^{-b} + V\beta_1)(n_k^{-b} + V\beta_1 + 1)} \tag{5.12}$$

5.4 Experimental Results

5.4.1 Dataset

Tweets 2011: It was published in TREC 2011 microblog track, which provides approximately 16 million tweets sampled between January 23rd and Februray 8th, 2011. It includes an user id, complete content of tweets, and a timestamp for each tweet. Several preprocessing steps used to improve the quality of text. Finally, the number of left tweets is 3,132,485, average document length is 6.31, and the vocabulary size is 83,129.

Sentiment 140: It contains 1,600,000 tweets extracted using the twitter api. The polarity of tweets have been annotated. The average document length is 9.13 with 115,209 distinct words.

Web snippets: It collects 12,340 web search snippets. We performed the following preprocessing on this dataset: 1) covert letters to lowercase, 2) remove all nonalphabetic characters and stop words. The average length of each snipper is 16.02 words. The corpus contains 6,632 words.

Statistics on the three datsets are reported in Table 5.1.

Table 5.1 Statistics on the Three Datasets

Dataset	*Docs*	*Average Document Length*	*Vocabulary Size*
Tweets 2011	3,132,485	6.31	83,129
Sentiment 140	1,600,000	9.13	115,209
Snippets	12,340	16.02	6,632

5.4.2 Baseline

We compare our proposed TN-BTMF against the following topic models.

• LDA [79] is one of the most classical topic models.

• BTM [18] learns topics by modeling the generation of word co-occurrence patterns, and works as the basis of our model.

• GPU-BTM [81] is designed to extend BTM by incorporating auxiliary information via GPU scheme.

Prior distribution parameters $\alpha = 50/K$ and $\beta = 0.01$ for all methods. We run Gibbs sampling for 1000 iterations. As for GPU-BTM and our proposed method TN-BTMF, a promotion μ is set as 0.3.

5.4.3 Word Embedding

In this work, FastText word embedding is used. It contains 300-dimensional word vectors trained on the Wikipedia 2017, UMBC webbase corpus and statmt.org new datsete. For GPU-BTM method, only the top 20 words with highest similarity are used.

5.4.4 Topic Coherence Evaluation

In this work, UCI topic coherence is chosen to evaluate topic models. UCI topic coherence uses point-wise mutual information to measure the coherence of topics. For a given topic z, top-N probable words are selected to calculate the average PMI score of each pair of these words:

$$PMI(z) = \frac{2}{N(N-1)} \sum_{1 \le i < j \le N} log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \tag{5.13}$$

where $p(w_i, w_j)$ is the probability that words w_i and w_j appear in the same document. The overall topic coherence for each model is the averaged PMI-Score over all learned topics. An external corpus is needed to calculate the PMI-Scores. In our experiment, we use 1 million English Wikipedia articles for 3 public datasets.

Figure 5.5 Figure 5.7 compares the topic coherence of all models on three public datasets with $T = 5, 10, 20$ and number of topics $K = 40, 60, 80$, respectively. For both datasets, the proposed method TN-BTMF achieves the best topic coherence across all settings. GPU-BTM is the second best model in most cases. When topics equal to 40, all models achieve the best coherence performance.

5.4.5 Model Complexity

LDA is a method to draw topic assignment for every word occurrence in document, which cost time $O(K|D|\bar{l})$, where $\bar{l} = \sum_i \frac{l_i}{|D|}$ is the average length of documents in the collection. The time complexity of BTM is $O(K|B|)$ for one iteration, where K is the number of latent topics and $|B|$ is the size of biterms. $|B| \approx \frac{|D|\bar{l}(\bar{l}-1)}{2}$. Extended with the GPU model, GPU-BTM has a time complexity of $O(K|B| + |B|\tau)$, where τ is a coefficient by considering the cost involved by applying the GPU model. The time complexity of TN-BTMF is $O(K|B| + |B|\tau + |D|\bar{l})$, where $|D|\bar{l}$ is the time complexity to detect noise biterm.

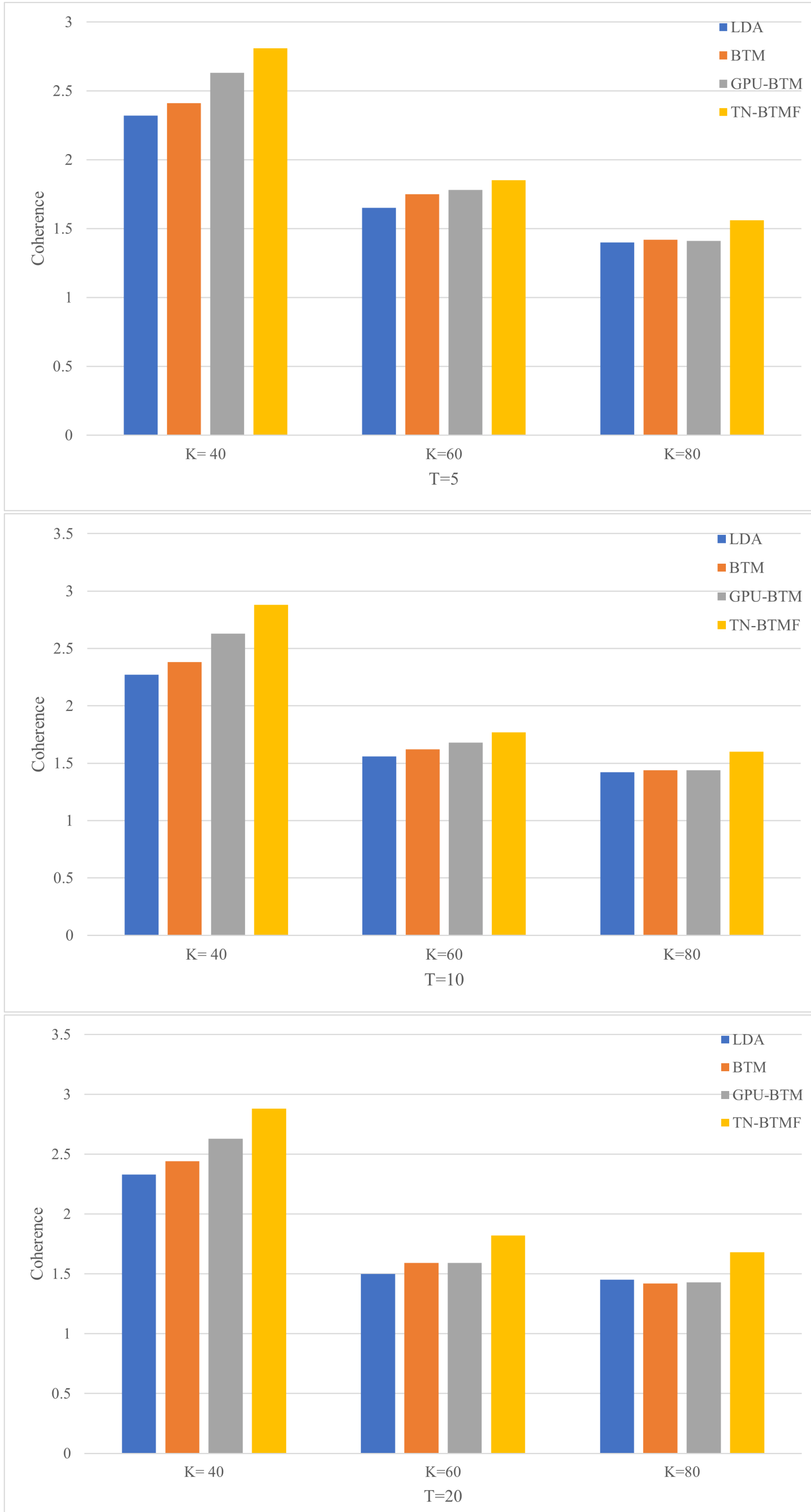

Figure 5.5 Topic coherence evaluated on Tweets 2011 dataset.

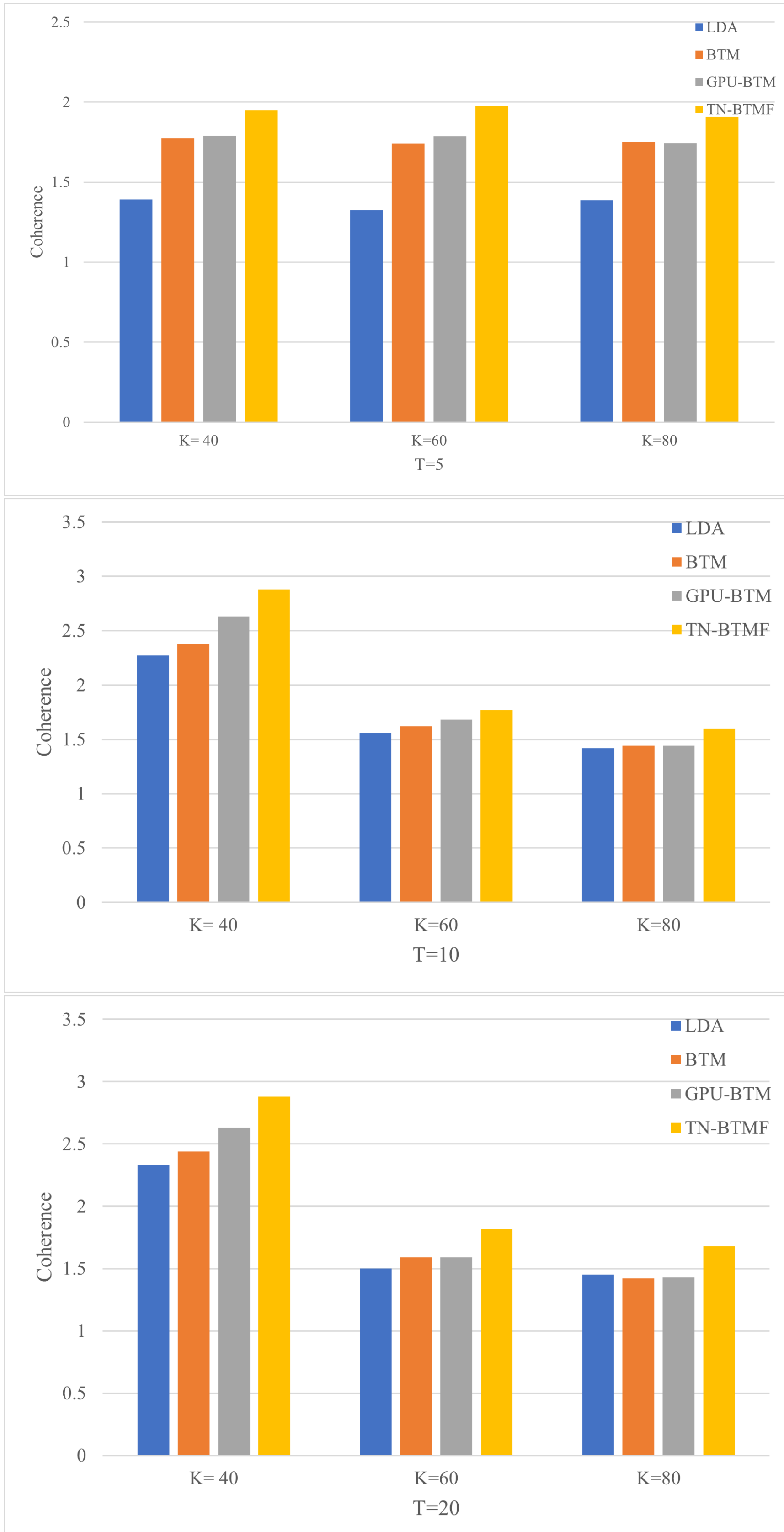

Figure 5.6 Topic coherence evaluated on Sentiment140 dataset.

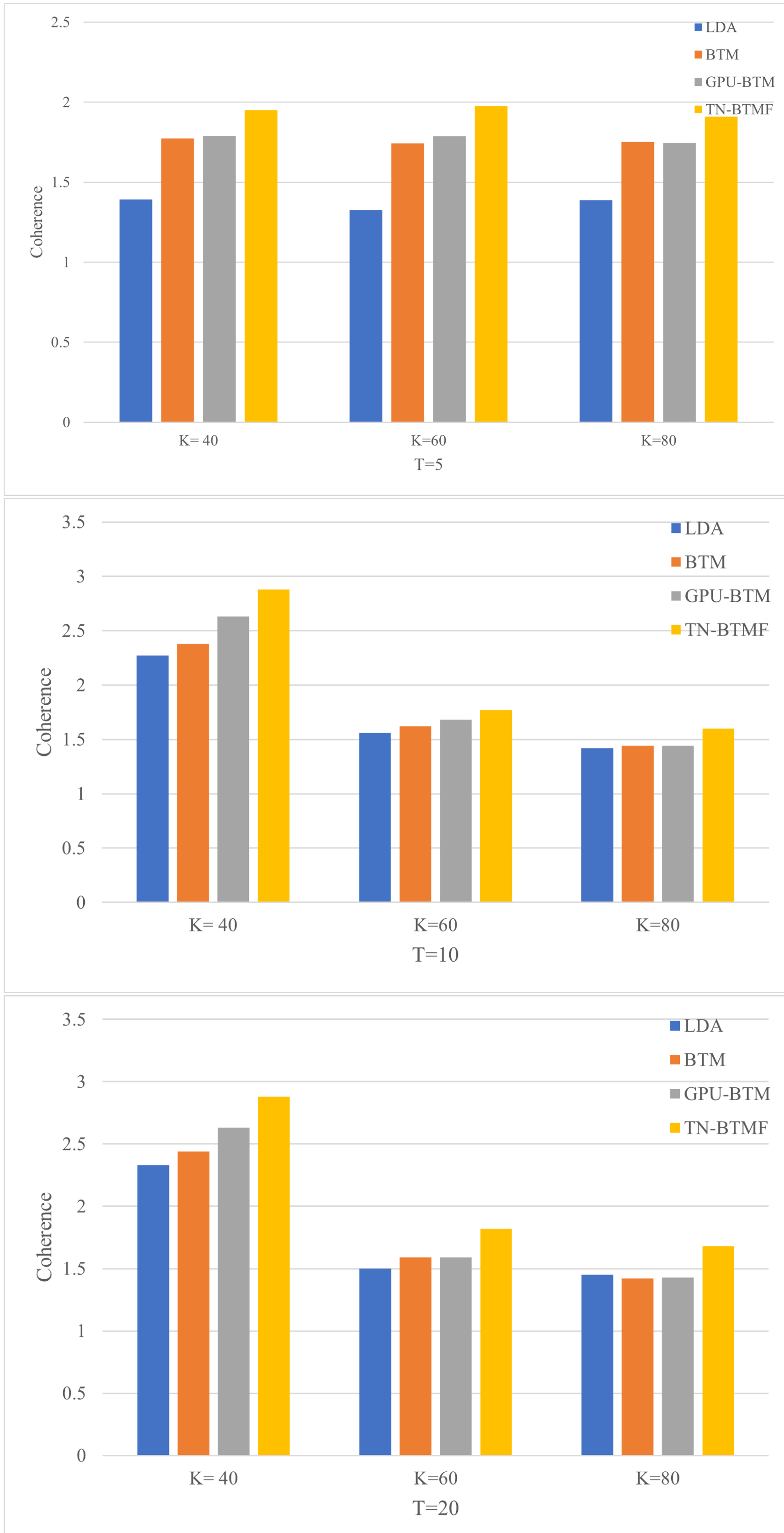

Figure 5.7 Topic coherence evaluated on Snippets dataset.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Summary of Contribution of This Dissertation

Social media, because of their prevalence and important, have attracted an increasing number of scholars to study its content. Among them, social media text is the most common and widest in scope at present, such as topic modeling, opinion mining, and sentiment classification. Based on the study of topic modeling methods applicable to social media text, this work focuses on the mining of COVID-19 vaccine related events and their topic evolution problems. The specific work and contributions can be summarized as follows:

This work aims at three core directions of social media analysis: 1) exploring the sentiment information of social events; 2) discovering hidden patterns among social events; and 3) dealing with sparsity problem of short text. Finding the sentiment information of social events can reflect the peoples' public opinions toward them and can be used to monitor public attitudes during their evolution. The second direction identifies people's interest in their discussion topics of social events by using a topic modeling method. The last direction focuses on the sparsity problem caused by the limited length of social media data.

Sentiment analysis is a method to analyze subjective texts to classify their sentiment into positive, negative, or neutral for a given target. It also determines the emotional preference tendency of the view of the text. A comparison study of aspect-based sentiment analysis is performed in this work. The aspect-based sentiment analysis methods are categorized into three classes: lexicon-based, machine learning, and deep learning methods. Three publicly available datasets are introduced to evaluate recently ABSA methods. It can be inferred that the joint information features, which are obtained during aspect extraction, are one of the useful features to improve the sentiment classification performance. Adding document-level labeled corpora is an effective way to enrich the training information, and then improving the classification performance.

Then, this work aims at discovering hidden patterns among social events in Chapter 4. A sentiment analysis and topic modeling framework is introduced to find the public opinions and interest trends among social events. The dataset includes more than one year tweets where the COVID-19 vaccine received the FDA approved EUA. The positive and negative polarity change curves demonstrate that initially only a minority of people had positive attitudes toward vaccines. Then a significant increase in overall positive attitudes as vaccine research progressed. 11 major topics are identified by using the framework.

Lastly, since the social media text has limited length and caused a sparsity problem, conventional topic models usually face the challenges of incoherent and unintelligible topic representation. In this work, a novel topic model, named TN-BTMF is proposed to alleviate this problem. It utilizes the FastText word embedding to bring new representation of words. A noise biterms detection method which combine words co-occurrence with semantic similarity is proposed to reduce noise. We conduct experiments on three real-world social media datasets. The experimental results show that TN-BTMF outperforms state-of-the art methods from topic coherence.

6.2 Limitations and Future Work

Social media-oriented topic analysis is a dynamic and challenging research area with many unresolved issues need to be further investigated.

The topics generated by the current work are only a list of words with high probability in that topic, and it is difficult to fully understand the topic if a user is not familiar with a concern social media dataset. Therefore, how to accurately explain the meaning of each topic is a problem to be addressed as future work.

In addition to considering sentiment analysis and topic modeling as two separate tasks. The topic modeling results can be treated as different aspects for sentiment analysis task. And then a aspect-based sentiment analysis method can be applied to achieve more accurate sentiment classification result.

In the study of topic modeling, although the proposed method has some advantages over several state-of-art methods, it is static. Therefore, dynamic or online learning model-based topic modeling method must be considered in future research [82] [83] [84] [85] [86]. The ability to use complex network analysis methods to predict the future evolutionary trends of topics is another direction for future work.

REFERENCES

[1] R. Murakami and B. Chakraborty, "Investigating the efficient use of word embedding with neural-topic models for interpretable topics from short texts," *Sensors*, vol. 22, no. 3, p. 852, 2022.

[2] F. Wu and Y. Huang, "Sentiment domain adaptation with multiple sources," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 301–310.

[3] E. Cambria, D. Das, S. Bandyopadhyay, and A. Feraco, "Affective computing and sentiment analysis," in *A Practical Guide to Sentiment Analysis*. Springer, 2017, pp. 1–10.

[4] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures on Human Language Technologies*, vol. 5, no. 1, pp. 1–167, 2012.

[5] H. Wang, D. Can, A. Kazemzadeh, F. Bar, and S. Narayanan, "A system for real-time twitter sentiment analysis of 2012 us presidential election cycle," in *Proceedings of the ACL 2012 System Demonstrations*, 2012, pp. 115–120.

[6] W. Yue, Z. Wang, J. Zhang, and X. Liu, "An overview of recommendation techniques and their applications in healthcare," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 4, pp. 701–717, 2021.

[7] X. Luo, W. Qin, A. Dong, K. Sedraoui, and M. Zhou, "Efficient and high-quality recommendations via momentum-incorporated parallel stochastic gradient descent-based learning," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 2, pp. 402–411, 2020.

[8] G. Wei, Q. Wu, and M. Zhou, "A hybrid probabilistic multiobjective evolutionary algorithm for commercial recommendation systems," *IEEE Transactions on Computational Social Systems*, vol. 8, no. 3, pp. 589–598, 2021.

[9] J. Qiang, Z. Qian, Y. Li, Y. Yuan, and X. Wu, "Short text topic modeling techniques, applications, and performance: a survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 34, no. 3, pp. 1427–1445, 2020.

[10] M. C. Phan, A. Sun, Y. Tay, J. Han, and C. Li, "Neupl: Attention-based semantic matching and pair-linking for entity disambiguation," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 1667–1676.

[11] J. Wang, L. Chen, L. Qin, and X. Wu, "Astm: An attentional segmentation based topic model for short texts," in *2018 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2018, pp. 577–586.

[12] B. Pang and L. Lee, "A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts," *arXiv preprint cs/0409058*, 2004.

[13] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004, pp. 168–177.

[14] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proceedings of the First Workshop on Social Media Analytics*, 2010, pp. 80–88.

[15] D. Alvarez-Melis and M. Saveski, "Topic modeling in twitter: Aggregating tweets by conversations," in *Tenth international AAAI Conference on Web and Social Media*, 2016.

[16] Y. Zuo, J. Wu, H. Zhang, H. Lin, F. Wang, K. Xu, and H. Xiong, "Topic modeling of short texts: A pseudo-document view," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 2105–2114.

[17] J. Yin and J. Wang, "A dirichlet multinomial mixture model-based approach for short text clustering," in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2014, pp. 233–242.

[18] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *Proceedings of the 22nd International Conference on World Wide Web*, 2013, pp. 1445–1456.

[19] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[20] R. Das, M. Zaheer, and C. Dyer, "Gaussian lda for topic models with word embeddings," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2015, pp. 795–804.

[21] C. Li, H. Wang, Z. Zhang, A. Sun, and Z. Ma, "Topic modeling for short texts with auxiliary word embeddings," in *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2016, pp. 165–174.

[22] T. H. Nguyen and K. Shirai, "Phrasernn: Phrase recursive neural network for aspect-based sentiment analysis," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 2509–2514.

[23] T. T. Thet, J.-C. Na, and C. S. Khoo, "Aspect-based sentiment analysis of movie reviews on discussion boards," *Journal of Information Science*, vol. 36, no. 6, pp. 823–848, 2010.

[24] L. Jiang, M. Yu, M. Zhou, X. Liu, and T. Zhao, "Target-dependent twitter sentiment classification," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011, pp. 151–160.

[25] S. Kiritchenko, X. Zhu, C. Cherry, and S. Mohammad, "Nrc-canada-2014: Detecting aspects and sentiment in customer reviews," in *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, 2014, pp. 437–442.

[26] Y. Jo and A. H. Oh, "Aspect and sentiment unification model for online review analysis," in *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, 2011, pp. 815–824.

[27] D. Kumar Gupta, K. Srikanth Reddy, A. Ekbal *et al.*, "Pso-asent: Feature selection using particle swarm optimization for aspect based sentiment analysis," in *International Conference on Applications of Natural Language to Information Systems*. Springer, 2015, pp. 220–233.

[28] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.

[29] J. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," 2001.

[30] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, "Semeval-2015 task 12: Aspect based sentiment analysis," in *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, 2015, pp. 486–495.

[31] D.-T. Vo and Y. Zhang, "Target-dependent twitter sentiment classification with rich automatic features," in *Twenty-fourth International Joint Conference on Artificial Intelligence*, 2015.

[32] F. Liu, T. Cohn, and T. Baldwin, "Recurrent entity networks with delayed memory update for targeted aspect-based sentiment analysis," *arXiv preprint arXiv:1804.11019*, 2018.

[33] A. Graves, "Long short-term memory," *Supervised Sequence Labelling with Recurrent Neural Networks*, pp. 37–45, 2012.

[34] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[35] R. Socher, C. C. Lin, C. Manning, and A. Y. Ng, "Parsing natural scenes and natural language with recursive neural networks," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, 2011, pp. 129–136.

[36] J. Weston, S. Chopra, and A. Bordes, "Memory networks," *arXiv preprint arXiv:1410.3916*, 2014.

[37] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *Journal of Machine Learning Research*, vol. 12, no. ARTICLE, pp. 2493–2537, 2011.

[38] Y. Zhang and B. Wallace, "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification," *arXiv preprint arXiv:1510.03820*, 2015.

[39] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[40] B. Huang, Y. Ou, and K. M. Carley, "Aspect level sentiment classification with attention-over-attention neural networks," in *International Conference on Social Computing, Behavioral-cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*. Springer, 2018, pp. 197–206.

[41] W. Xue and T. Li, "Aspect based sentiment analysis with gated convolutional networks," *arXiv preprint arXiv:1805.07043*, 2018.

[42] X. Li, L. Bing, W. Lam, and B. Shi, "Transformation networks for target-oriented sentiment classification," *arXiv preprint arXiv:1805.01086*, 2018.

[43] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "An interactive multi-task learning network for end-to-end aspect-based sentiment analysis," *arXiv preprint arXiv:1906.06906*, 2019.

[44] A. Kumar, V. T. Narapareddy, V. A. Srikanth, L. B. M. Neti, and A. Malapati, "Aspect-based sentiment classification using interactive gated convolutional network," *IEEE Access*, vol. 8, pp. 22 445–22 453, 2020.

[45] M. Zhang, Y. Zhang, and D.-T. Vo, "Gated neural networks for targeted sentiment analysis," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

[46] J. Cheng, S. Zhao, J. Zhang, I. King, X. Zhang, and H. Wang, "Aspect-level sentiment classification with heat (hierarchical attention) network," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 97–106.

[47] D. Tang, B. Qin, and T. Liu, "Aspect level sentiment classification with deep memory network," *arXiv preprint arXiv:1605.08900*, 2016.

[48] S. Ruder, P. Ghaffari, and J. G. Breslin, "A hierarchical model of reviews for aspect-based sentiment analysis," *arXiv preprint arXiv:1609.02745*, 2016.

[49] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based lstm for aspect-level sentiment classification," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 606–615.

[50] J. Liu and Y. Zhang, "Attention modeling for targeted sentiment," in *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 2017, pp. 572–577.

[51] D. Ma, S. Li, X. Zhang, and H. Wang, "Interactive attention networks for aspect-level sentiment classification," *arXiv preprint arXiv:1709.00893*, 2017.

[52] F. Fan, Y. Feng, and D. Zhao, "Multi-grained attention network for aspect-level sentiment classification," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 3433–3442.

[53] Y. Tay, L. A. Tuan, and S. C. Hui, "Dyadic memory networks for aspect-based sentiment analysis," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 2017, pp. 107–116.

[54] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier, "Exploiting document knowledge for aspect-level sentiment classification," *arXiv preprint arXiv:1806.04346*, 2018.

[55] D. Hazarika, S. Poria, P. Vij, G. Krishnamurthy, E. Cambria, and R. Zimmermann, "Modeling inter-aspect dependencies for aspect-based sentiment analysis," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018, pp. 266–270.

[56] D. Yin, X. Liu, X. Wu, and B. Chang, "A soft label strategy for target-level sentiment classification," in *Proceedings of the Tenth Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2019, pp. 6–15.

[57] M. Hu, S. Zhao, L. Zhang, K. Cai, Z. Su, R. Cheng, and X. Shen, "Can: Constrained attention networks for multi-aspect sentiment analysis," *arXiv preprint arXiv:1812.10735*, 2018.

[58] L. Bao, P. Lambert, and T. Badia, "Attention and lexicon regularized lstm for aspect-based sentiment analysis," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, 2019, pp. 253–259.

[59] D. Tang, B. Qin, X. Feng, T. Liu *et al.*, "Target-dependent sentiment classification with long short term memory," *arXiv preprint arXiv:1512.01100*, 2015.

[60] W. Gao, M. Peng, H. Wang, Y. Zhang, Q. Xie, and G. Tian, "Incorporating word embeddings into topic modeling of short text," *Knowledge and Information Systems*, vol. 61, no. 2, pp. 1123–1145, 2019.

[61] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural Networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[62] Y. Tay, L. A. Tuan, and S. C. Hui, "Learning to attend via word-aspect associative fusion for aspect-based sentiment analysis," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[63] H. T. Nguyen and M. Le Nguyen, "Effective attention networks for aspect-level sentiment classification," in *2018 10th International Conference on Knowledge and Systems Engineering (KSE)*. IEEE, 2018, pp. 25–30.

[64] L. Dong, F. Wei, C. Tan, D. Tang, M. Zhou, and K. Xu, "Adaptive recursive neural network for target-dependent twitter sentiment classification," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (volume 2: Short papers)*, 2014, pp. 49–54.

[65] W. Wang, S. J. Pan, D. Dahlmeier, and X. Xiao, "Recursive neural conditional random fields for aspect-based sentiment analysis," *arXiv preprint arXiv:1603.06679*, 2016.

[66] P. Chen, Z. Sun, L. Bing, and W. Yang, "Recurrent attention network on memory for aspect sentiment analysis," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 452–461.

[67] Q. Liu, H. Zhang, Y. Zeng, Z. Huang, and Z. Wu, "Content attention model for aspect based sentiment analysis," in *Proceedings of the 2018 World Wide Web Conference*, 2018, pp. 1023–1032.

[68] R. Ma, K. Wang, T. Qiu, A. K. Sangaiah, D. Lin, and H. B. Liaqat, "Feature-based compositing memory networks for aspect-based sentiment classification in social internet of things," *Future Generation Computer Systems*, vol. 92, pp. 879–888, 2019.

[69] S. Sukhbaatar, J. Weston, R. Fergus *et al.*, "End-to-end memory networks," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[70] S. Wang, S. Mazumder, B. Liu, M. Zhou, and Y. Chang, "Target-sensitive memory networks for aspect sentiment classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018.

[71] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. Al-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq *et al.*, "Semeval-2016 task 5: Aspect based sentiment analysis," in *International Workshop on Semantic Evaluation*, 2016, pp. 19–30.

[72] M. Saeidi, G. Bouchard, M. Liakata, and S. Riedel, "Sentihood: Targeted aspect based sentiment analysis dataset for urban neighbourhoods," *arXiv preprint arXiv:1610.03771*, 2016.

[73] M. Mitchell, J. Aguilar, T. Wilson, and B. Van Durme, "Open domain targeted sentiment," in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 1643–1654.

[74] L. Deng and J. Wiebe, "Mpqa 3.0: An entity/event-level sentiment corpus," in *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015, pp. 1323–1328.

[75] M. A. Ferrag, L. Shu, and K.-K. R. Choo, "Fighting covid-19 and future pandemics with the internet of things: Security and privacy perspectives," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 9, pp. 1477–1499, 2021.

[76] E. F. Ohata, G. M. Bezerra, J. V. S. das Chagas, A. V. L. Neto, A. B. Albuquerque, V. H. C. de Albuquerque, and P. P. Reboucas Filho, "Automatic detection of covid-19 infection using chest x-ray images through transfer learning," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 1, pp. 239–248, 2020.

[77] R. Gallotti, F. Valle, N. Castaldo, P. Sacco, and M. De Domenico, “Assessing the risks of ‘infodemics’ in response to covid-19 epidemics,” *Nature Human Behaviour*, vol. 4, no. 12, pp. 1285–1293, 2020.

[78] C. Hutto and E. Gilbert, “Vader: A parsimonious rule-based model for sentiment analysis of social media text,” in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 8, no. 1, 2014, pp. 216–225.

[79] D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation,” *Journal of Machine Learning Research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[80] R. Churchill and L. Singh, “Topic-noise models: modeling topic and noise distributions in social media post collections,” in *2021 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2021, pp. 71–80.

[81] Y. Guo, Y. Huang, Y. Ding, S. Qi, X. Wang, and Q. Liao, “Gpu-btm: A topic model for short text using auxiliary information,” in *2020 IEEE Fifth International Conference on Data Science in Cyberspace (DSC)*. IEEE, 2020, pp. 198–205.

[82] X. Hong, T. Zhang, Z. Cui, and J. Yang, “Variational gridded graph convolution network for node classification,” *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 10, pp. 1697–1708, 2021.

[83] L. Hu, S. Yang, X. Luo, and M. Zhou, “An algorithm of inductively identifying clusters from attributed graphs,” *IEEE Transactions on Big Data*, 2020.

[84] X. Liu, M. Yan, L. Deng, G. Li, X. Ye, and D. Fan, “Sampling methods for efficient training of graph convolutional networks: A survey,” *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 2, pp. 205–234, 2021.

[85] H. Wu, X. Luo, M. Zhou, M. J. Rawa, K. Sedraoui, and A. Albeshri, “A pid-incorporated latent factorization of tensors approach to dynamically weighted directed network analysis,” *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 3, pp. 533–546, 2021.

[86] Z. Zhao, H. Zhou, L. Qi, L. Chang, and M. Zhou, “Inductive representation learning via cnn for partially-unseen attributed networks,” *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 1, pp. 695–706, 2021.