COMBINED WORD AND NETWORK EMBEDDINGS : AN ANALYSIS FRAMEWORK OF USER OPINIONS ON SOCIAL MEDIA

by

Tannu Dharmendra Singh

A thesis submitted to the faculty of
The University of North Carolina at Charlotte
in partial fulfillment of the requirements
for the degree of Master of Science in
Computer Science

Charlotte

2020

Approved by:

____
Dr. Siddharth Krishnan

____
Dr. Samira Shaikh

____
Dr. Gabriel Terejanu

ABSTRACT

TANNU DHARMENDRA SINGH . Combined Word and Network Embeddings : An Analysis Framework of User Opinions on Social Media. (Under the direction of DR. SIDDHARTH KRISHNAN)

Online social media like Twitter, Facebook, and Gab is often used as the stage to deliver one's opinion for a particular group of people, a political party, etc. Sometimes, the opinions shared are considered as controversial by some audience, applauded by some, or disagreed by some in the form of comments, sharing, likes, or dislikes. The information about shared opinion and the reaction to it in the form of positive, or negative reaction forms an interaction, and a collection of many such interactions forms a signed network. In addition, the evolution of information on social networks strongly relies on the nature of interactions between the users. The study of interactions is, therefore, crucial to predict the extent and nature of information spread. In this work, we study the relationship between users whether they agree or disagree in the dynamic evolution of interactions (cascades) on a larger network, Gab, to predict the relationship between the users on the social network. We quantitatively use the combination of text information and network information to enhance state of the art deep learning models for contradiction detection. The outcome of this research might contribute to improving link prediction.

DEDICATION

Dedicated to my family.

ACKNOWLEDGEMENTS

First, I would like to thank my advisor, Dr. Siddharth Krishnan for giving me this opportunity and providing his support and guidance throughout the completion of my thesis. I would also like to thank my thesis committee: Dr. Samira Shaikh and Dr. Gabriel Terejanu for their constant support, feedback, and encouragement.

Great appreciation to the University of North Carolina at Charlotte for providing me with necessary tools, space and aiding me with assistantship in the entire process. I would also like to thank my parents Dharmendra Singh, Sunita Singh, my sisters Swati Singh, Neha Singh, and my friend Kanishk Chauhan for their constant encouragement. I would like to thank Dr. Arunkumar Bagavathi and my fellow Pedram Bashiri at Network Analytics and Social Computing Lab (NASCL) for helping me with their research insights, and valuable time.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI An acronym for Artificial Intelligence.

ASNE An acronym for Attributed Social Network Embedding.

BERT An acronym for Bidirectional Encoder Representations from Transformers.

Bi-LSTM An acronym for Bi-directional Long Short Term Memory.

CENE An acronym for content-enhanced network embedding.

ESIM An acronym for Enhanced Sequential Inference Model.

LINE An acronym for Large-scale Information Network Embedding.

LSA An acronym for Latent Semantic Analysis.

LSTM An acronym for Long short-term memory.

MMDW An acronym for max-margin DeepWalk.

NE An acronym for Notwork Embedding.

NetS An acronym for net score.

NLI An acronym for Natural Language Interference.

NLP An acronym for Natural Language Processing.

RNN An acronym for Recurrent Neural Network.

SCGN An acronym for Signed Graph Convolutional Networks.

TADW An acronym for Text- associated matrix factorization DeepWalk.

CHAPTER 1: INTRODUCTION

In our research, we use Gab data as the social network to study the user interactions and their behaviour. Gab was launched in August 2016 which garnered a lot of support during US Presidential elections. Gab promotes free speech by allowing users to share their opinion and broadcast it to all the users on Gab. It has caught attention from media to politicians when one of the users Robert Browers on Gab was involved in Pittsburgh Synagogue shooting who used to broadcast his anti-semitic views. The reason we are using Gab data is because the posts or comments on Gab are not moderated unlike other social media like Twitter, Reddit, or Facebook. Because of this, users usually tend to agree or disagree with one another, which makes this platform ideal for our research. Gab is a social media site that calls itself the champion of free speech. The site does not prohibit a user from posting any hateful content. This led us to focus our study on Gab (Gab.ai). Also, to understand the true nature of the interactions between users, we need to study them in an environment that would not stop them from following/enacting their beliefs. The demonstration of the content being non moderated is shown in fig 1.1. The recent research is studying the interaction between users on social media has been done in many ways such as, using the text information to predict the sentiment, using the network information to predict the link between users, detecting communities of like-minded users, detecting fake news and detecting opinions to name a few. The text associated with the users on social media and the network behavior such as how many people interact to each other, and who replies to whom forms a real world data to perform the social network analysis. The text information could be processed using Natural Langugae Processing (NLP) which aims to interpret the text as intended while the the network information could be represented and processed using network embedding. Only a few research focuses on utilizing the text information and the network information to study the social media. We believe that the network information and the text information along the network are correlated to each other and when combined mimics the real world social media data.

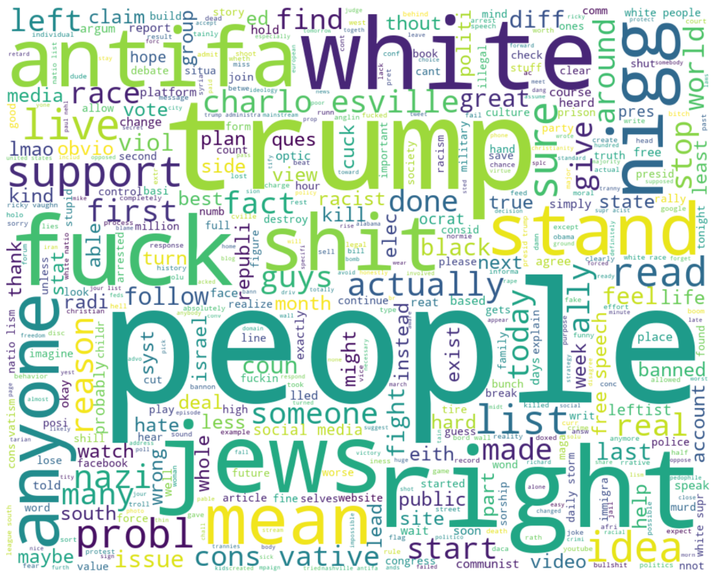

Figure 1.1: Word Cloud of aggregated content from posts from Gab.ai

1.1 Problem Statement

With the data we have and network properties that we formed, we address the following problem:

Predicting the sign of an edge $E_{ij} \epsilon (V, E)$ in a cascade based on previous interactions and edge attributes of the cascade by combining the text embedding (te) and network embedding (ne) at time (t) while considering that the number of nodes (V) remain the same for a given network, N, where $(V, E) \epsilon N$. The text associated with the network N changes overtime as the network evolves and so does the edges formed between the nodes V. The weight of the edges at a given time could be determined by normalizing the value of agreement to disagreement ratio in range [0,1] between the nodes (users). After adding the weight, we could label the edges (signed interactions) between two nodes as +1, or -1 where +1 represents agreement between two users in an interaction while -1 represents disagreement. A small illustration of such a network of users represented as nodes and the connection between them represented as edges is shown in fig1.2.

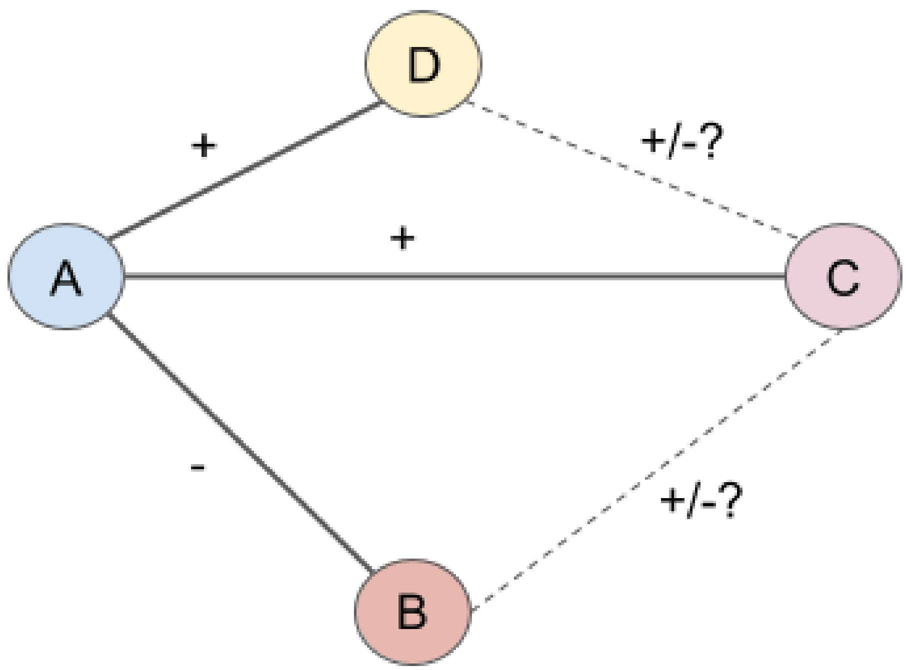

Figure 1.2: This figure represents a small network of users having signed edges between them represented by solid lines while, the possible edges (represented by sign) between two non-connected nodes is represented by dashed line.

1.2 Motivation

The information flow on social media unveils a lot of crucial information regarding people's opinion, their sentiment, and also helps to determine communities in the social network to name a few. Social media has become a crucial part of any marketing and advertising business due to its wide popularity and acceptance in the mass. Studying relationships between user interaction on social media helps to study information flow, popularity of users, and studying their sentiment. Social media platforms like Twitter, Facebook, and Reddit have garnered millions of users to use their platform and share their opinions. Due to the millions of diverse users on social media some users might agree or disagree with other people opinions. Such kind of arguments between users forms a signed network. Although multiple type of relations can be represented by signed networks with positive and negative edges, but the widespread of social media has found the increasing need to represent the relationship between users in the form of signed network. This prevalence of signed network motivated us to enhance the existing way to study signed network and the arguments between the users on social media. In this research we aim to study the information flow in the form of signs (agreement or disagreement) on social media by enhancing an existing model which incorporates the topology of the conversations on social media and also the textual information through of the conversations.

1.2.1 Importance of studying social media

Since the inception of social media it has been widely used as a platform to express, and promote opinions. Many social media like facebook, and Twitter has their own analytical platform which helps them to identify the problems associated with the platform and they use this information to improve the platform such as recommending friends to users, or improving the platform to improve the user experience. Similarly many business leaders use the information present on social media to know the sentiment about their product usage and the acceptance of product in general public. For instance, Amazon improves their customer services by mining through the comment network and user profile network to recommend the best products to similar users, and to identify the best buyers using the sentiments from user comments. In a nutshell, whenever a platform emerges where many people are interacting through messages, posts , comments, likes, and dislikes it makes an ideal information to do sentiment analysis, community detection, recommendation system, and link prediction. Scientist use these information available through social media to create a simulation, or machine learning model to predict the information flow, sentiment, stance, and popularity.

1.2.2 Ways to study social media

Social media analysis often leads to applications opinion mining, political affiliation, sentiment analysis, bot detection, cyber criminal detection, and hate speech detection etc. There are many ways to use social media information like using the textual information to study the sentiments, and hate speech detection, using the topology information to study the link prediction, community detection, and using the topology and textual information to study stance detection, hate speech detection, and community detection. There are other ways to study and develop the theory of social media done by social scientist.

1.2.3 Contribution

Our contribution in this research are different ways to combine the text and topology information, and then adding it to the input layer of the existing state of the art model. We have also labeled the data using the compound score difference of each post by users in an interaction and then defining a threshold value to classify the labels, which in our case are 'agreement' and disagreement. We eventually compared the performance of the existing state of the art model against the enhanced models that we have adapted on Gab preprocessed dataset.

1.3 Thesis Outline

This section gives the brief outline of all the chapter following this chapter. Chapter 2 briefly describes all the major components and the background concepts we used in our base model and all the enhanced models. Chapter 3 briefly overviews the related work in network embeddings, signed network embedding, and stance detection. In chapter 4 we described the detailed functioning and architecture of our base model and enhanced models. We wrap the methodology by showing results and performance comparison of all the models in chapter 5. In the last chapter we conclude the findings in this research and also mention future work which could extend our enhanced models.

CHAPTER 2: BACKGROUND

In this section we explained all the major methods used in the enhanced models that we used to predict the agreement or disagreement in an interaction. We have introduced Natural Language Interference (NLI) because one of the the state of the art models ESIM [2] is based on NLI. We also described the network embedding and word embedding process in detail. The working of LSTM is described in detail because it the main component of Bi-LSTM (Bi-directional Long Short Term Memory) model which is used in ESIM model.

2.1 Natural Language Inference

Since artificial intelligence emerged, inference has always been the prime area of research and development, but there has been inadequate research on the topic of NLI. For instance, understanding whether a plausible inference could be made from natural language premise p to natural language hypothesis h. The main issue of NLI lies in semantic understanding, lexical and semantic knowledge, the process of logical thinking, and varying linguistic properties.

An example of a hypothesis and premise is mentioned below where 'p' represents the premise and 'h' represents the hypothesis. Based on this example, the inference could be estimated as contradictory because hypothesis 'h' implies the opposite of that of 'p' when we estimate the semantic value of p and h. Estimating the label of inference such as contradictory, non-contradictory, or neutral is a difficult difficult task due to the sheer similarity between two contradictory hypothesis and premise.

p - A man is on duty and inspecting the road.

h - A man inspecting the road is littering.

While NLI entails identifying the skewed relation of inferability between a premise p and hypothesis h, it could be also extended to the task of identifying a consistent relation of between p and h in the form of semantic equivalence, which suggest, we can identify semantic equivalence of p and h given that we have a system which is capable of discovering whether a plausible inference could be made from natural language premise p to natural language hypothesis h and by using this system in bidirectional way. The simple way to determine the rough semantic equivalence between texts or words is to use manually constructed dictionary such as WordNet [3]. Although the ability to determine whether two texts or sentences have similar words but used in different context is comprised using the dictionary method. To overcome this problem NLI was introduced which helps to determine the inferability or semantic equivalence between sentences precisely as intended.

2.2 Word Embedding

Word embedding is used to represent a learned representation of text such that, the words that that have same meaning have a matching representation. This way of representing textual information which is easily adapted by machine learning model due to the vectorized representation is considered one of the major advancements in the field of natural language processing. Word embedding is the technique to represent each single word in a text as a real-valued vectors in a predefined vector space. Due to the way of representing each word in text to a single vector and these set of vector values are learned in a way such that it mimics neural networks eventually, leading to the use of these vector values into deep learning models. Usage of words determine the distributed representation of vectors. This results in similar representation of words that are used in similar ways eventually capturing their real definition. In contrast the bag of words model represents different words have different representation irrespective of how they are used and unless explicitly handled. This problem could be solved using distributed representation for each word.

During the word embedding process, each word is represented by a real valued vector which is usually more than hundreds of dimensions. While during the one-hot embedding word representations are usually in millions of dimensions to represent the sparse word representation.

A small example of how words which are used in similar ways have the similar representation is shown in fig 2.1. In this example, the text or conversation between two people is represented on a two dimensional plane and the words like discussion and meeting are close to each other because they are used in similar ways.

Figure 2.1: This figure represents the how word embedding works when a bunch of text corpus is converted to word embeddings. In this figure, an example of text from two people A, and B has been represented into word embeddings.

2.2.1 Word Embedding Algorithms

The process of word embedding from a large corpus of text is done by learning a real-valued vector representation for a predefined fixed size dictionary of words.

The learning process involves incorporating neural network for text prediction, or documnent classification, or sentiment analysis while in some cases it is also incorporated with unsupervised process such as creating document statistics. The following methods are used to learn text embedding from text corpus.

2.2.1.1 Embedding Layer

Creating word embedding using the embedding layer is most suitable when it is used in conjunction with a deep learning model or a neural network model where the task is specified such as document classification, text classification, and predicting the sentiment.

The prerequisite of using this method of word embedding is that the text corpus should be cleaned and preprocessed in such a way that each word is represented as one-hot encoding. The fixed dimension of the vectorized form of these embeddings could be described as the segment of the model that is used, for instance, the dimensions could be 50,100, or 300 or as specified. The initial values of the vectors are filled with random numbers. The embedding layer is usually defined in the input encoding of a neural network model and it is fitted to the model using the backpropagation algorithm[4].

The input to this embedding layer one-hot encoded words is projected to the vectors representing the words. This embedding formed can be used in varying neural network architecture, for example, the vectors formed using this embedding method needs to be combined or added together before being used in the multilayer perceptron model. For neural network models like a recurrent neural network, the vectors formed using this embedding method could be directly used representing each word as a single vector into the input encoding. Creating word vectors using the embedding layer involves a lot of training data which impacts the runtime of the model and thus leads to a slow process of training and predicting.

2.2.1.2 Word2vec

Word2Vec[5] is an embedding method where the word embedding is generated from a text corpus using statistical and mathematical concepts. This method does not require a lot of training data to train the embedding model thus making this method of embedding efficient for neural network training. The following models adapt the embeddding approach of Word2vec:

- Continuous Skip-Gram Model.
- Bag-of-Words, or CBOW model

.

The CBOW[5] model adapts the learning of embedding from word2vec by forecasting the existing word based on the context it is being used. On the other hand, the continuous skip-gram model learns the embedding by forecasting the neighboring words when we have the current word. It could be derived that both the models involve the contextual knowledge of the current word and the contextual knowledge and use of the neighboring words. Mostly, the contextual knowledge is locally derived from the neighboring words by defining the window of how many words to consider as the neighboring words. The window is mutable and it changes as per the defined configuration of the model. One of the major benefits of using the window technique is that it requires less memory and it is simple to implement further producing the great quality of word embeddings. This method also allows for a higher dimension of word embeddings which needs to be formed from a huge text corpus (more than hundreds of billions of words).

2.2.1.3 GloVe

Global Vectors for Word Representation (GloVe) [6], extends the word2vec embedding algorithm by improving the learning process of representing word vectors. The conventional way to represent vector representation of words was done using matrix factorization techniques, for example, Latent Semantic Analysis (LSA) which uses statistics of global text but the performance modern method like word2vec and gloVe outperforms these conventional methods of representing texts. The modern way of representing word vectors ensures that the true meaning of the word is captured and applying the results on the prediction and classification tasks of the neural network model, an example demonstrating the working of gloVe embedding is shown in fig **??**.

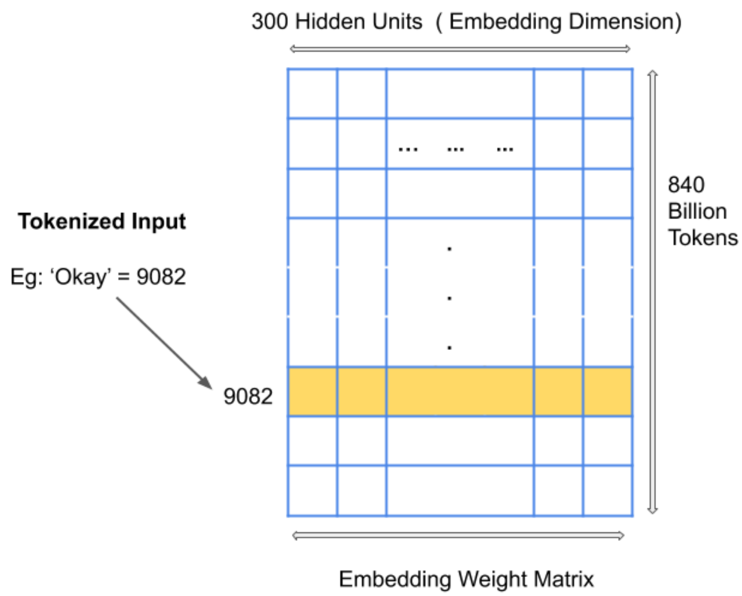

Figure 2.2: This figure represents how gloVe embedding uses matrix factorization. In the figure we encoded the word 'okay' as integer 9082. Then to get hidden layer output value for âokayâ we just simply need to lookup the 9082nd row in the weight matrix.The number of dimension in the hidden layer output is the embedding dimension

GLoVe embedding combines the benefit of the local context of words from word2vec and the matrix factorization technique from the conventional embedding methods like LSA. This combination of different ways of word embedding process gives the gloVe embedding method and edge when the performance in terms of time and space is compared to the existing methods of word embeddings. To implement the window technique from word2vec gloVe assemble distinct word context or word co-occurrence matrix using global statistics of words across the text corpus. Hence, this word embedding method results in a better and efficient representation of words.

2.3 Network Embedding

The most common way to represent social media user networks is by using graphs or networks. Out of the many application of network representation like connections in protein molecules, and connections of neurons in the brain the network representation of social media is leading the research in deep learning. Presenting the social network in the form of graphs seems to be powerful and emulates the real-time connection of users. Applying machine learning techniques on networks could be beneficial to many machine learning task predictions and their variation. To elaborate it, the link prediction could be solved when a machine learning uses data in the form of a network, it will help to predict whether a link could be formed between two separate individuals.

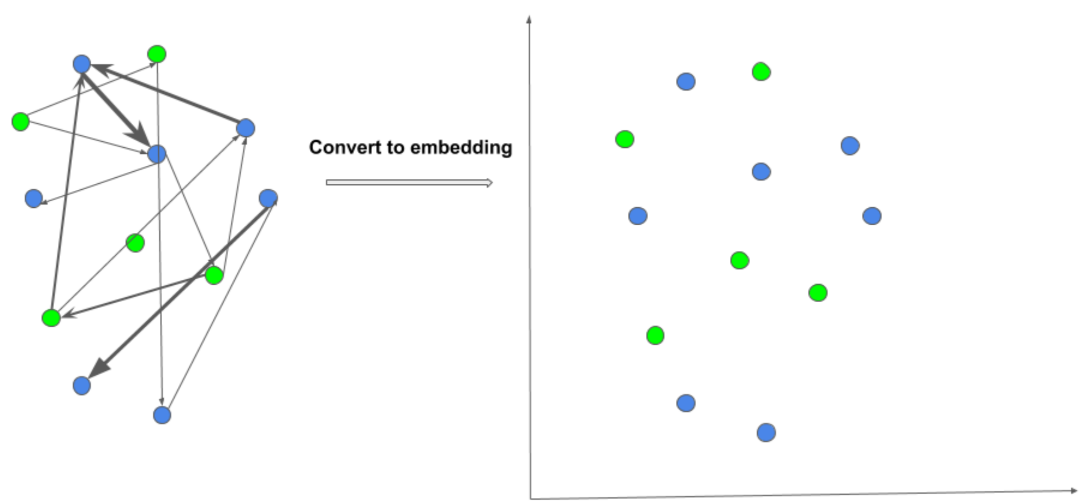

Figure 2.3: This figure represents the simple way a complex network with varying edge weights between two nodes in a network could be represented into low-dimensional vector space.

As discussed above in the word embedding section, machine learning models work well when the input is represented in the vectorized form which aligns with the neural network architecture. This leads to the formation of network embedding from networks, where the network is represented into low level vectorized form and the node which is similar or closer to each other is also represented similarly. The weight on the edges of the network used to represent the learned vector representation of the network. The fig 2.3 represents the simple way a complex network with varying edge weights between two nodes in a network could be represented into a low-dimensional vector space.

2.4 Long short term memory (LSTM)

The major shortcoming with conventional neural networks is that they are unable to understand the context of the sequential data because they do not have the memory to process the previous sequence of data. For every pattern of data traditional neural network learns from scratch even though if the pattern is repeating multiple times. LSTM is a type of advanced RNN which overcomes the problem of storing the pattern in the memory by having a loop in the network which allows information to stay in the network. Also, the traditional neural network had the problem of vanishing and exploding gradient but these problem has been also overcome by LSTM. Vanishing gradient occurs while training the model when the error form the back-propagation seems to exist and effect the last layer of the neural network. Exploding gradients also has a similar effect but instead of effecting the last layer, it affects the weights of the neural network when an error occurs which seem to manipulate the whole system of neural network. Tackling the exploding gradient is comparatively easier than identifying the vanishing gradient. LSTM effectively solves the problem of vanishing gradient by enabling the neural network to apprehend the longer dependencies between the layers. When the backpropagation algorithm giving the prime importance to weights which leads to larger weight values. Further, this could lead to inconsistent values for the weights which results in an unstable neural network.

This leads to an unstable network. On the other hand, the vanishing gradient arises commonly when the activation has a very small gradient. When the backpropagation algorithm allows the multiplication of the weights and the low gradient, it leads to the weights being vanished completely as proceed to the final layer. The varying process of the backpropagation algorithm discussed above causes the traditional neural network to forget the long term dependency. The vanishing gradient problem could be identified by using ReLU activation function, or by modifying the identity matrix by initializing it with the weights of the network. LSTM gives the option to choose varying activation functions which also includes sigmoid and tanh, eventually helping to identify the vanishing gradient problem.

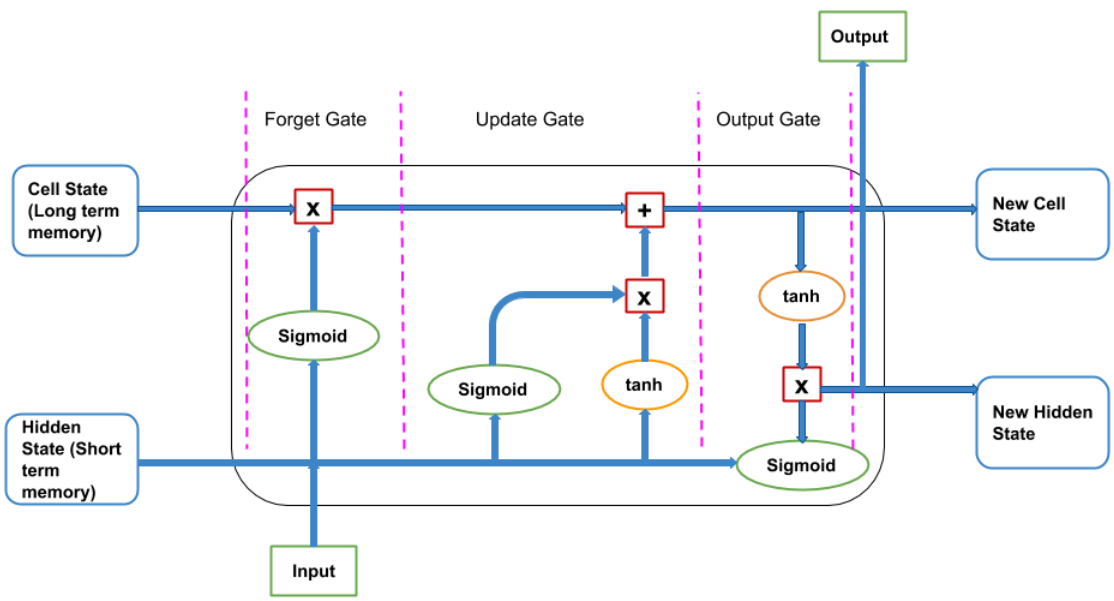

Figure 2.4: This figure represents the architecture of long short term memory (LSTM).

LSTM is an enhanced version of RNN. The detailed architecture of LSTM is shown in fig 2.4. The cell state in LSTM stores information and is a crucial part of LSTM while enabling to store or reject a piece of information. The sigmoid layer and the pointwise multiplication are the gates in the LSTM which also helps to decide whether to store or get rid of information. LSTM constitutes three major elements update, forget, output gate. The forget gate consist of the cell state, and the sigmoid layer, cell state helps in storing the information while the sigmoid helps in the decision making process. Forget gate determines whether the cell state should keep or get rid of any information. This decision of forget state is regulated by the sigmoid layer in the forget gate. The sigmoid and tanh layer helps in generating new candidate value which updates the cell state if the decision to store the information has been processed through the sigmoid layer. The final gate called the output gate determines which chunks of information from the cell state go to the output.

2.5 Text Classification

Text classification is the process of allocating a set of predefined categories to the text data. Text classifiers have multiple applications like organizing, restructuring, summarizing, and categorizing the text. It is also used for binary and multiclass classification of text. For instance, conversations on social media could be categorized by topics, a large set of documents could be classified into various categories, and the trending topic on social media could be identified. The popular way to do text classification is by using manual and automatic text classification methods. Manual text classification does not involve any computing efforts rather it is mostly done by humans by directly interpreting the text and labeling them. Human involvement in classification is time-consuming so most of the research involves the text classification which is done computationally or automatically. The automatic text classification uses modern methods like machine learning, NLP, and deep learning to automate the process of classifying the text which is faster than manual classification.

2.6 Homophily Effect

People with different characteristicsâgenders, races, ethnicities, ages, class backgrounds, educational attainment, etc.appear to have very different qualities. The interpersonal network consists of homogeneous information. These qualities usually help in categorizing people into a group. For example, children are innocent, men and women are hardworking, and people involved in crimes are violent. These general attributes do not take into consideration the other attributes to classify people into different groups. Stating the fact that people generally only have major connections to people like themselves. This type of connection causes the homophily effect where people who behave similarly in different social situations are interacting with each other the most.

The general rule of homophily is that people who are similar are most likely to contact each other than the people who are not similar. This results in the localized information between similar people intact within the group so the information flow tends to be localized. Homophily also implies that that gap in terms of the attributes transform into a network gap, the number of attributes through which information should travel between two people. The homophily effect is used in many research such as contradiction detection, stance detection, and finding the relationship between users on social media in the form of agreement and disagreement.

2.7 Link Prediction

Link prediction is one of the most crucial research topics in the field of graph theory and networks. The main objective of link prediction is to identify whether a link could be formed between a pair of nodes which are connected indirectly or not connected at all. In social networks this method is used to predict the user interactions in the future to elaborate more, this concept is used to detect any kind of relationship between two users who are currently not connected to each other directly. The structured information extracted from a graph having node, edges, and features can be adapted to machine learning algorithms to predict the link between two nodes in a graph. The general applications of link prediction include predicting the future relations between users in a hateful network, the relationship between molecules in a protein, determining co-authors in the citation network, to name just a few.

2.8 Signed Network

The relationship users on social media sites could be perceived as negative or positive given they interact and endorse or oppose one another view. This type of relationship when applied on large scale social media users forms a signed network. Studying the signed network has emerged as an interesting field in social media theory. It could be possible that two users do not always have a negative or positive relationship between them, it might vary on the wide ranges of topics, for some of them they might endorse each other or oppose each other. This complex relationship between users makes it difficult to predict the sign or relationship between them with only a few interactions between them. It might be difficult to predict a relationship between two users in terms of the sign if a new topic of discussion happens between them.

CHAPTER 3: RELATED WORK

For network analysis of large social networks, studying the network embedding has been a crucial part of the research. In recent years, there have been a large number of network embedding (NE) models proposed to learn efficient node embeddings [7], [8], and [9]. For instance, [10] does random walks over networks and introduces a learning model which aligns with the word representation , Skip-Gram [5], to learn network embeddings. LINE [7] effectively help in the optimization of conditional and joint probabilities of edges in complex and large networks like social networks to learn node representation, sometimes called vertex representation. Node2vec [11] enhances the way to explore network structure efficiently by modifying the random walk strategy in DeepWalk into biased random walks. To summarise, most of the above mentioned NE models only use the encoded structural information into node embeddings, while excluding the heterogeneous information or attributed information accompanied with nodes which does not result to emulate real-time social network information. Representing the heterogeneous information along the nodes and edges might result in better prediction model when we are trying to solve a problem on social media. To shed the light on this crucial heterogeneous information, many researchers make substantial efforts to integrate it into the traditional NE models. For example, [8] proposed text-associated Deep-Walk (TADW) to boost matrix factorization based DeepWalk with textual information. [9] present max-margin DeepWalk(MMDW) to learn distinguished network representations by making use of labeling information of nodes. [2] present group enhanced network embedding (GENE) to consolidate existing group information in the NE. [12] propose an embedding which considers textual information as a special kind of nodes, eventually leading to a context-enhanced network embedding (CENE) through leveraging both topological or structural and textural information to learn network embeddings. Atttributed social network embedding (ASNE)[13] learns node representation by using structural as well as attribute proximity. The recent research in combining the text and network embedding has been done in [14], which combines these embeddings considering that the number of nodes remains the same but the text associated with it changes overtime.

On contrary, for text analysis of large social networks, studying the text embedding has been a crucial part of the research. In recent years there have been a large number of proposed ways to incorporate text emeddings into deep learning model. One such example is COBW [5] which can train billions of words for unlimited size vocabulary. Another model which is widely used for this purpose is ESIM where it uses NLI to predict the inferability between a hypothesis and a premise.

Signed network was initially used determine the concept of balance in graphs [15]. Early work in signed network also indicates the formation of the concept of balance in triangles of sentiments and the concept of balance in signed graphs [16]. One of the recent studies in signed network devises a way to study various features which aligns with balancing the graph [17]. Signed networks have been studied wide ranges of context. For instance, [18], and [19] study signed graphs where the edges are directed and developed a theory to question the importance of nodes in such networks. Other areas of research explore edge and node classification [20], [21], link prediction [22], [23], community detection [24], [25], [26], [27], and recommendation [28]. A thorough survey of signed network is done in [29]. The another couple of research explore the problem of detecting hostile communities in signed networks [30]. These work shows the research on undirected signed network [31]. [32] which uses the concept of local optima. Signed Graph Convolutional Networks (SCGN) [33] proposed a framework that utilizes balance theory to appropriately combine and use the information across layers of a SGCN model.

There has been a few work which uses stance detection but not involving the social media users [34], [35]. Detecting stance uses the linguistic properties and social interactions between the users [36]. Most of the previous studies define stance as a textual entailment task where the main processing depends on the raw text only [37], [38], [39], [40], [41]. In this way of stance detection, a given hypothesis is inferred from the premise. In this research it has been concluded that constructing a knowledge based dataset about the give topic aids in stance detection task[39]. This helps in using the knowledge based dataset in stance detection task to set of predefined topics. Sometimes the topic cannot be determined from a text. To analyse such kind of text one can use the stance detection to find any contradiction in the text. For instance, [42] defined a list of keywords that identifies the stance between the two politician (Trump and Hillary). Using this defined keywords it was easier to identify the undetected stance towards the politician (Trump). Another study [41] extends the same construction of keywords that contains words that are contracdictory and non-contracdictory for each labels or categories. Similarly, [43] used a domain dataset related to Trump along with lexicon to construct a labeled dataset to identify stance towards Trump. Further, [44] constructed author embedding from users tweets to predict the stance. There are some work which focuses on network and content information without taking into consideration about the political views [45], [35], [46], [47]. For instance the study of [35] helps to identify liberals and conservative.

It could be easily derived from the previous work of NE that all existing NE models treat text not as a part of network while creating a network embedding while all text based models also do not consider any structural information of a network into consideration. Similarly we can also say that about signed networks where more network embedding approach are used instead of combining both topological and text information. The stance detection models discussed above mainly uses the text information associated with the social media. In contrast, we propose a deep learning model which uses text and network information both in the embedding, but including text as the node attribute, to predict the agreement and disagreement. Recent advancement in network embedding indicates which combines heterogeneous textual information along the network increases the model efficiency to predict real time social network problems.

CHAPTER 4: METHODOLOGY

In this chapter we present the dataset description, data preprocessing, data labeling, base model, and enhanced models. We describe the number of users and interactions we considered for our signed network analysis, we labeled the dataset using NLTK [1] and compound score difference, then we evaluated the labeled dataset using net score and dabatepedia dataset, finally ended the data preprocessing section by combining the network and word embeddings for the models. We introduce the base model and all the enhanced models which considers both the network and word embedding in the input encoding layer. The models that we have used is based on Bi-LSTM and uses homophilly concept. All the text attributed networks where nodes are represented using text information such as social network, and co-author network represent the homophilly effect. This concept also applies to our work where we aim to predict the sign between two nodes in a signed network of interactions on social media.

4.1 Dataset Description and Preprocessing

4.1.1 Dataset

After filtering out all the posts which did not have any comments we extracted $\approx 21M$ posts spanning around two years since the inception of the site. We classified these posts into 1.8M cascades as described in [48] which has information about the structure of the interactions on Gab. The total number of interactions in all the cascades[48] is 21 million. We used these interactions from the cascade as the input to the enhanced models.

Each row in the dataset consists of interactions in the cascade [48]. Additionally,

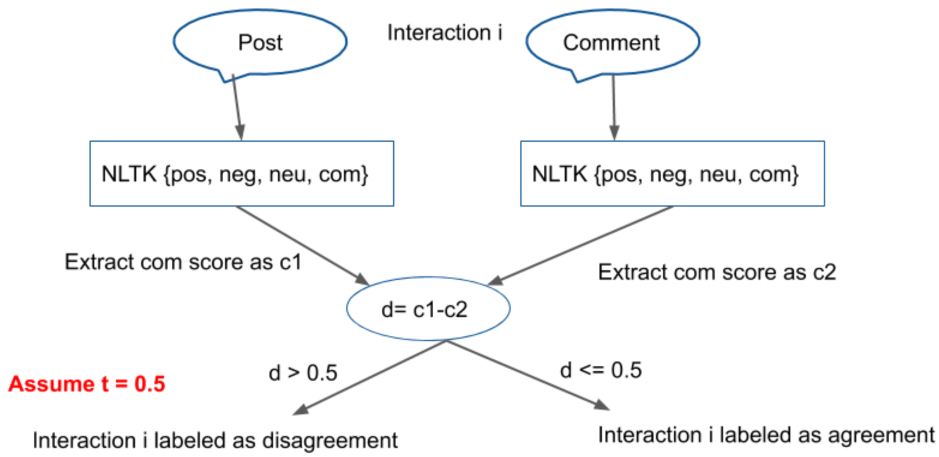

Figure 4.1: This figure shows the process of dataset labeling we have performed for each interaction in the network. We assumed a threshold value t = 0.5 which is compare against the difference d in the compound score of each post in an interaction. The difference of posts in an interaction d > 0.5 then that interaction was labeled as disagreement, or agreement otherwise.

we unpacked all the hashtags into words and appended it to the end of each of the posts in an interaction to which it belonged. Since the dataset that we used is raw and unlabeled data from Gab.ai, we computed the sentiment score of each post (node in cascade) which ranges from [-1,1], using NLTK VADER (Valence Aware Dictionary and Sentiment Reasoner), a lexicon, and rule-based sentiment analysis tool[1].

To extract the sentiment score of a post from an interaction we used compound score as the parameter. Then we computed the difference between the sentiment score of each post in the cascade to classify the interaction as agreement or disagreement. The threshold to classify an interaction was set to 0.5, meaning an interaction having a difference in compound score greater than 0.5 was classified as disagreement, and agreement otherwise. Using this threshold value we got 225596 agreements in the network and 961686 disagreements which resulted in 1:4 agreement to disagreement.

We also calculated the difference between positive score and negative score of each

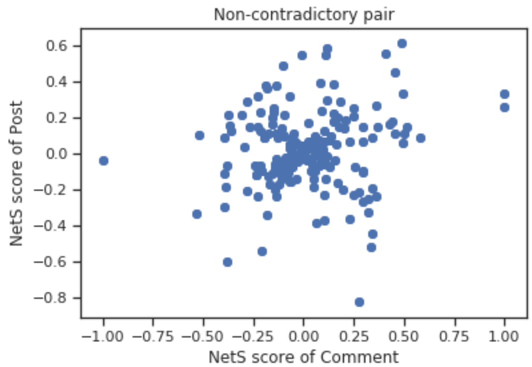

Figure 4.2: NetS of interactions labeled as " agreement". For a non-contradictory pair of interaction where users agree, the net score which is the summation of positive score and negative score from NLTK Vader[1] will tend towards a positive value. In this figure after randomly sampling 1000 interactions, we can see that the NetS value of non-contradictory pair of interaction is concentrated towards a positive value >0.0.

post in an interaction called net score (NetS) similar to the one calculated in [49]. The better NetS value implied the sentence to be positive, or negative otherwise. We evaluated our labeling method by plotting the NetS value of all the pairs labeled as an agreement (non- contradictory), and all the pairs labeled as disagreement (contradictory) in fig **??**, and fig **??**. The other way we evaluated our labeling method was by labeling the debatepedia dataset[50] using our method and comparing the labels. After comparing the labels using our NLTK [1] compound score difference method against the original labels we found that our proposed method gave an accuracy of 69.6%. This accuracy on labeled data motivated us to use this labeling method and the labeled dataset for predicting the sign of an edge between two nodes in an interaction.

To classify the interactions based on compound score difference we assumed the threshold value to be 0.5. This threshold value classified the interactions into unbalanced labels of agreements and disagreement. This led us to test various threshold value t and its effect on the number of labels in the dataset which in our case is agreement and disagreement.W e got a varying number of agreements and disagreements when we changed the threshold value. The effect of threshold value on the number

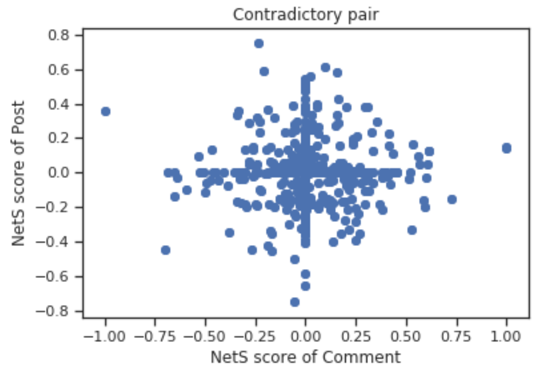

Figure 4.3: Net score of interactions labeled as "disagreement". For a contradictory pair of interaction where users disagree, the net score which is the summation of positive score and negative score from NLTK Vader [1] will tend towards a negative value. In this figure after randomly sampling 1000 interactions, we can see that the net score value of non-contradictory pair of interaction is concentrated towards a positive value <0.0.

of labels is shown in fig4.4, where the difference in the number of agreements and disagreements is less when t is 0.6. Out of all of the t values, we plotted the top three best t values corresponding to the balanced labels. So, the t value of 0.6 gave the balanced dataset where the difference between the number of agreement and disagreement labels showed to decrease by almost half when compared to the threshold value of 0.5 and 0.4 generated labels.

4.1.2 Combined Embeddings

We also created text embeddings from the text in the interactions from cascades using a pre-trained global representation of words gloVe [6]. One of the prevalent methods to represent complex social networks is to use network embedding(NE) to represent it into low dimensional vectors where along with the network some information about the nodes and edges is also used. Network embedding is a method that embeds the nodes on the network into a vector space similar to the way any two nodes are close to each other. In the past most of the network embeddings are well suited for the unsigned network but, in our case, we have social network information where users sometimes agree or disagree which makes our network a signed network. The

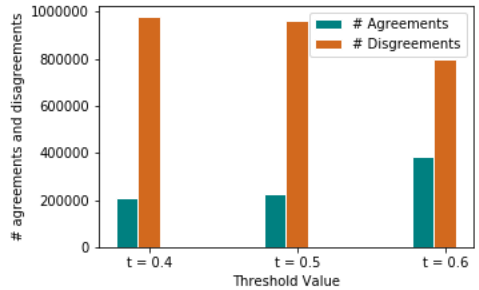

Figure 4.4: Effect of the threshold value to label the interaction. On the x-axis, we used the threshold value 't' which is the difference in the compound score of each post in an interaction, that we used to classify the interaction as agreement or disagreement. On the y-axis we used the total number of agreements and disagreements label against a specific threshold value.

network preprocessing step involved was to convert the signed network of interactions into low-level network embedding using Node2Vec[51] and ASNE[13].

The challenge lies in combining the text and network embeddings to use it in the input encoding of the deep learning model. We have adopted the combination of text and network embedding from [14] where we are just adapting one of the assumptions that the number of nodes remains the same in the network while the text related to the node can change over time. This method is useful in our research because of the similar nature of the problem with the number of nodes. Additionally, this method will also help in predicting the link formation between two users in a network at any point in time. The dynamic nature of text in this research also helps us present the arguments and agreements between users in a cascade of interaction effectively. This approach is better than network embedding because the network structure and the attributive information could be mutually supplemental. Finally, this method uses a dynamic text attribute network embedding method, which embeds nodes and words in a blended way, while considering timely changes in the vector representation of text and the nodes in the network.

The main idea of our method is based on the following three assumptions:

- connected nodes have similar representations on embedded space
- words associated with a node have similar representation on embedded space
- vector representations of nodes and words do not change dramatically among adjacent time segments.

4.2 Models Used

Since deep learning models have widely accepted and found to be effective in the better understanding semantic knowledge, we adopted one of the state-of-the-art deep learning models Enhanced Sequential Inference Model (ESIM) from [2], which proved to be effective in predicting contradicting pairs of sentences which are also depicted by the accuracy it provides. On top of this model, we experiment with different methods of incorporating network embedding and text embedding as input to the models.

We use Enhanced LSTM model[2] as our baseline model. We used this model [2] and tried to enhance it by adding a sentiment feature similar to [49], and aggregated text and network embedding (Node2Vec [51]). We also experimented with a different threshold value which was previously used to label agreement and disagreement. We have also used Attributed Social Network Embedding[13], which preserves the structural and attribute proximity of the network along with text embedding in our experiments to get the best result for our problem. We try to answer the following questions using the variations, and the feature sets:

1. Given a network of interactions between users, can we design the best model to predict the relationship between users in an interaction whether they agree or disagree?
2. Does the combination of text and network embedding help in improving the performance of the model?

3. How does the combination of features perform over the state of the art models existing to predict the relationship between two users?

4.2.1 Base Model

The important component of ESIM is a bidirectional LSTM model [?], which is called BiLSTM. BiLSTM was utilized twice in this ESIM model to capture the word order information and the sentence pair comparison information. This model is based on natural language inference[2] networks which are composed of the following major components: input encoding, local inference modeling, and inference composition. Here BiLSTM helps to represent a word and its context. We have also used BiLSTM to perform inference composition to construct the final prediction, where BiLSTM encodes local inference information and its interaction. While modeling a sequence, an LSTM employs a set of soft gates together with a memory cell to control message flows, resulting in effective modeling of tracking long-distance information/dependencies in a sequence. A bidirectional LSTM runs a forward and backward LSTM on a sequence starting from the left and the right end, respectively. The hidden state output from these two LSTMs is concatenated at each time step to form the context of each word.

The model was initialized with the 300D GloVe, 840B tokens pre-trained word vectors [6]. The OOV (out-of-vocabulary) words were randomly initialized. The dropout rate was set to 0.1, sequence length to 25, and the embedding's trainable parameter as true during the training. The Adam optimizer [?] was used during the model training. In our research, the goal of this model is to predict the relationship between two posts in interaction from the Gab dataset in the form of agreement or disagreement. The basic architecture of the base model is shown in **??**.

4.2.2 ESIM model with Node2vec

We tried to enhance the ESIM model by adding graph embedding to the input layer of the ESIM model along with glove [6] embedding. While word Embedding is the

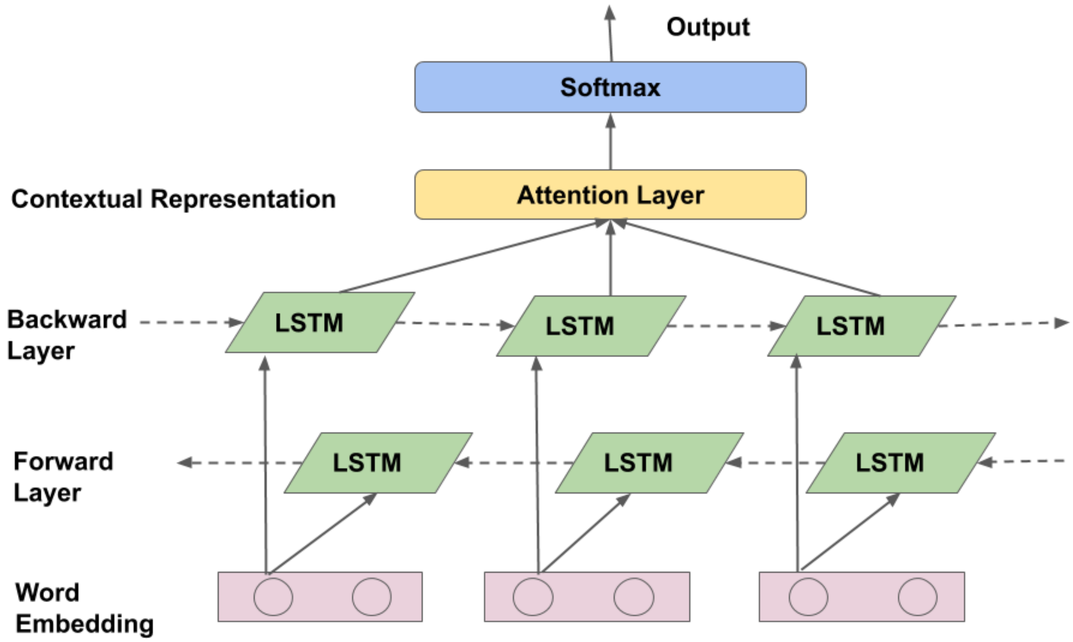

Figure 4.5: Enhanced LSTM for Natural Language Inference. Fig 4.5 is the baseline model to find agreement and disagreement in users interaction because it has been proved to be one of the best models for contradiction detection. This model uses Bi-LSTM to get the contextual representation and predicts the class of contradiction which in our cases are agreement, and disagreement.

process of representing words as dense vectors in a low-dimensional space [?], graph embeddings like node2vec [11] can overcome the downside of the sequential input. Word graph paths can connect semantically similar words or they can align similar words closer, which cannot be seen in many low-level connections. Those connections can be utilized for enhancing the word embedding. Hence, a network-based model can consider both the topological and the semantic information represented by text. The node2vec [11] framework learns low-dimensional representations for nodes in a network by optimizing neighborhood proximity. It uses the DeepWalk to go through the network but the random walks in node2vec are biased. Node2vec also aligns with the concept of homophily effect and structural balance. The demonstration of how a graph embedding like node2vec works with text embedding and modeled into BiLSTM model is shown in ??.

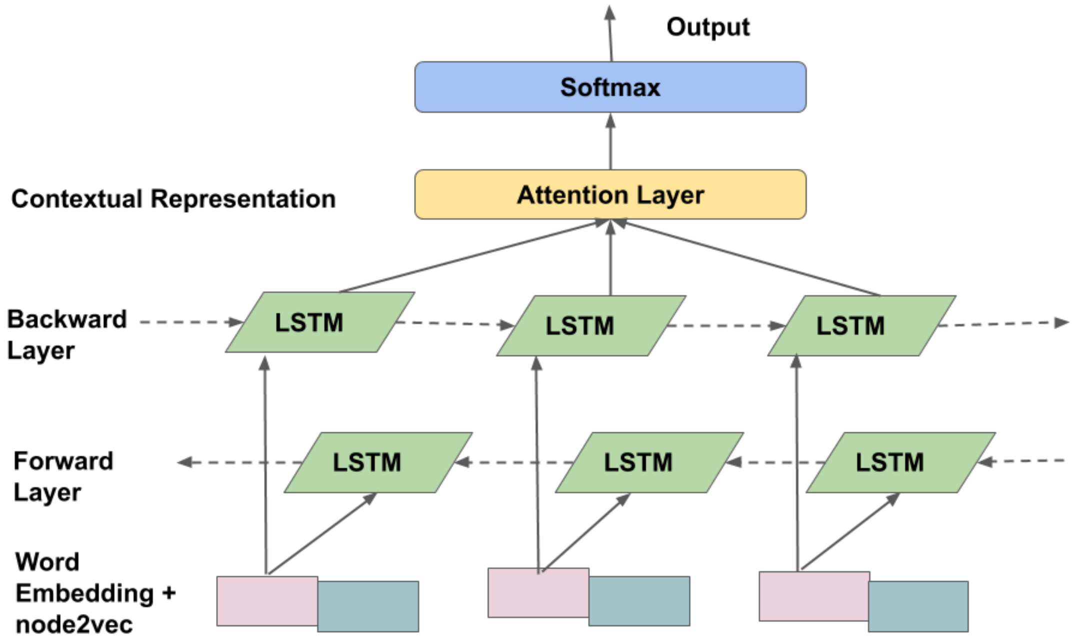

Figure 4.6: Enhanced LSTM for Natural Language Inference with node2vec. Fig 4.6 represents the architecture of the enhanced ESIM model. We tried to enhance it by adding node2vec along with glove embedding in the input layer.

4.2.3 ESIM model with ASNE

The another way we tried to enhance the ESIM model is by adding ASNE [13] embedding to the input layer of the ESIM model along with glove [6] embedding. In the previous approach with node2vec as the graph embedding we were not able to embed the attributed information along the nodes and the edges. The attributes in our network is the agreement and disagreement in between two users stored on the edges, and the textual information about the user's post present as node attribute. We are not considering any other user features in our model to maintain user confidentiality on Gab. The another motivation for considering ASNE is due to the attribute proximity in the embedding method aligns with the attribute homophily, which plays an important role in attribute related process. The demontration of how ASNE works with text embedding and modeled into BiLSTM model is shown in 4.7.

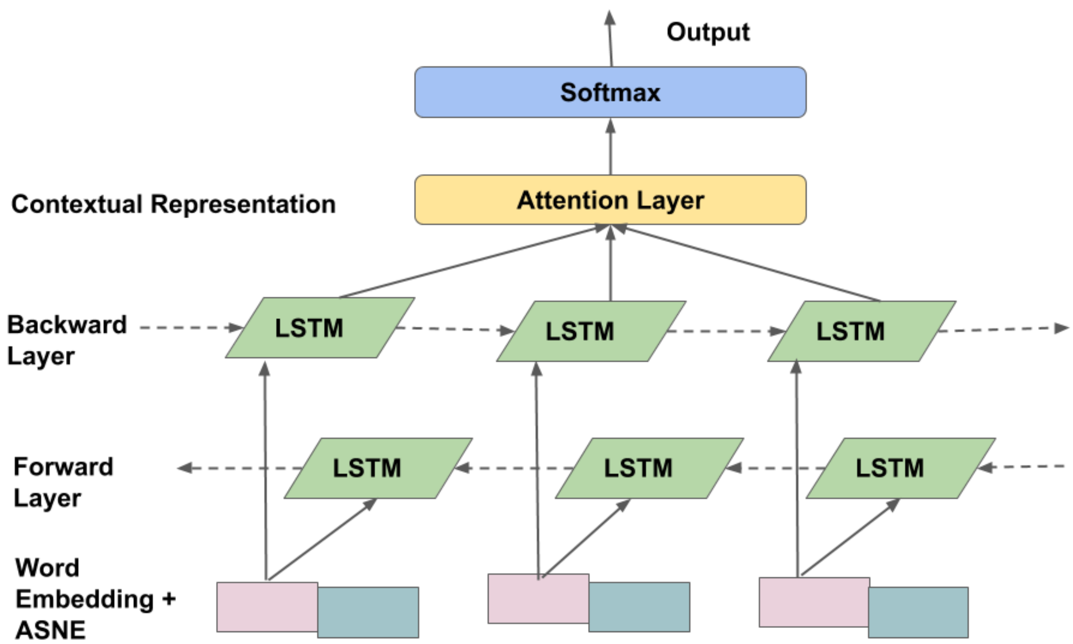

Figure 4.7: Enhanced LSTM for Natural Language Inference with ASNE.Fig 4.7 represents the architecture of another enhanced ESIM model. We tried to enhance it by adding ASNE along with glove embedding in the input layer.

Table 4.1: Performace of the enhanced ESIM models. We compared the enhanced model which has both text embedding and network embedding in the input layer of the model. In the table, 't' is the threshold value of difference in the compound score that we used to classify the interaction as agreement or disagreement. ESIM model is the base model , ESIM + Node2Vec is the model with node2vec also attached to ESIM input layer, and ESIM + ASNE is the model with ASNE also attached to ESIM input layer.

Model Description	F1-micro	Accuracy	Precision	Recall
ESIM (t= 0.04)	0.5982	0.6598	0.6956	0.5472
ESIM (t= 0.05)	0.6041	0.6751	0.7803	0.5467
ESIM (t= 0.06)	0.6257	0.6789	0.7402	0.5266
ESIM + Node2Vec (t = 0.04)	0.5848	0.6356	0.7034	0.5415
ESIM + Node2Vec (t = 0.05)	0.5989	0.6412	0.7156	0.5619
ESIM + Node2Vec (t = 0.06)	0.6204	0.6657	0.7374	0.5809
ESIM + ASNE (t = 0.04)	0.6021	0.6745	0.7754	0.5438
ESIM + ASNE (t = 0.05)	0.6145	0.6569	0.7275	0.5773
ESIM + ASNE (t = 0.06)	0.6334	0.7043	0.7289	0.5755

CHAPTER 5: RESULTS

In this study, we present a different approach to predict the relationship between user interactions on social media in the form of agreement or disagreement. The whole process of predicting the agreement and disagreement consisted of many small components that impacted the accuracy and F1 score of the models that we have used. To summarize, we extracted all the text information related to the user such as the user's post and hashtags and combined them into a single text blob, then we used the existing method [14] to combine the dynamic text and static node embeddings into a single embedding, we also experimented with various network embedding models which have proven to be effective in the link prediction task and combined these network embeddings with gloVe embedding. We first used the ESIM [2] model as a baseline and modified this model to accommodate the various combination of network and text embedding in the input encoding. We have reported the F1 score ESIM model, and the enhanced models in 4.1.

We innovatively integrate ESIM model with embedding features and evaluate the models by comparing them on the same dataset and same parameters. It is evident that with better network embedding methods like ESIM+ASNE on the same dataset when compared to ESIM and ESIM+Node2Vec this model gives the best F1-score. The threshold value 't' which is the difference in the compound score of each post in an interaction, that we used to classify the interaction as agreement or disagreement, also plays an important role in determining the performance of the model, because it changes the total number of agreement and disagreement labels in the dataset. The threshold score of 0.6 performs the best in all the models because the number of agreement and disagreement labels is more balanced when compared to the threshold

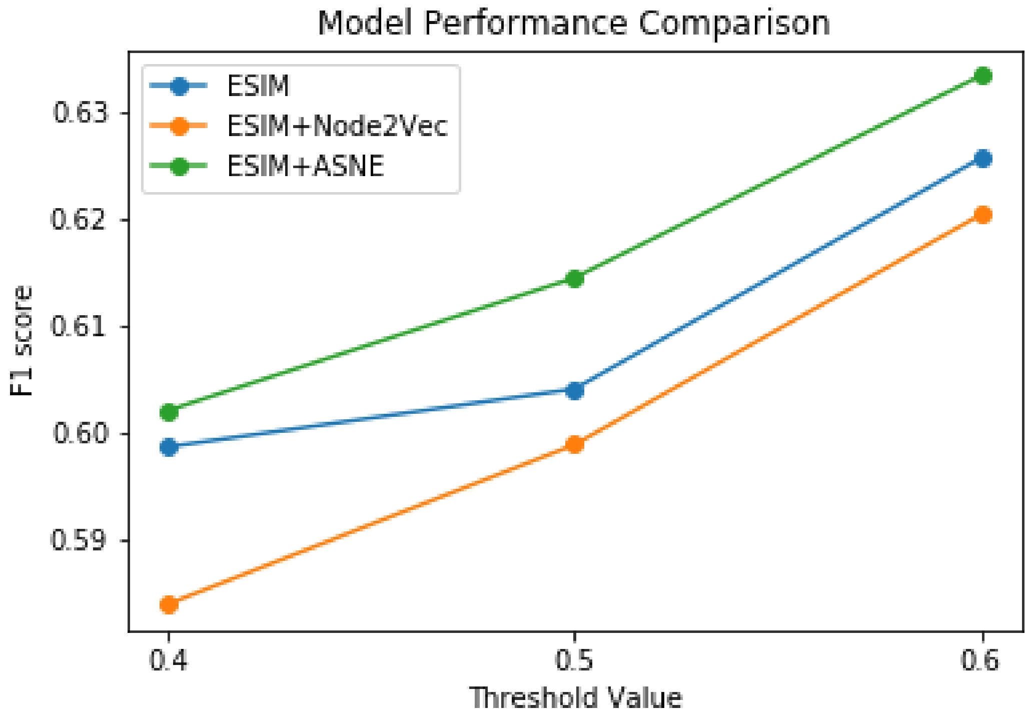

Figure 5.1: Model performance comparison.On the x-axis we used the threshold value 't' is compared against difference in the compound score of each post in an interaction, that we used to classify the interaction as agreement or disagreement. On y-axis we used the F1 score of ESIM, and each of the enhanced ESIM models.

value of 0.4 and 0.5 as shown in 5.1.

The effect of threshold value on the total number of agreements and disagreements in the dataset while labeling the could be seen in 4.4. In 4.4, it could be observed that when the threshold value is 0.4 the total number of labels assigned as agreement decreased making it more unbalanced as compared to the statistics when t is 0.5. The dataset where the interactions are labeled using t as 0.6 yields the best balanced dataset when compared to other t values. Through fig5.2 it could be implied that as the number of epochs increases the accuracy of the model increases until the number of epochs reaches 300 after that there was a decline in the accuracy. Similarly, in fig5.3 it is evident that the loss decreases with an increase in the number of epochs until the epoch value reaches 300.

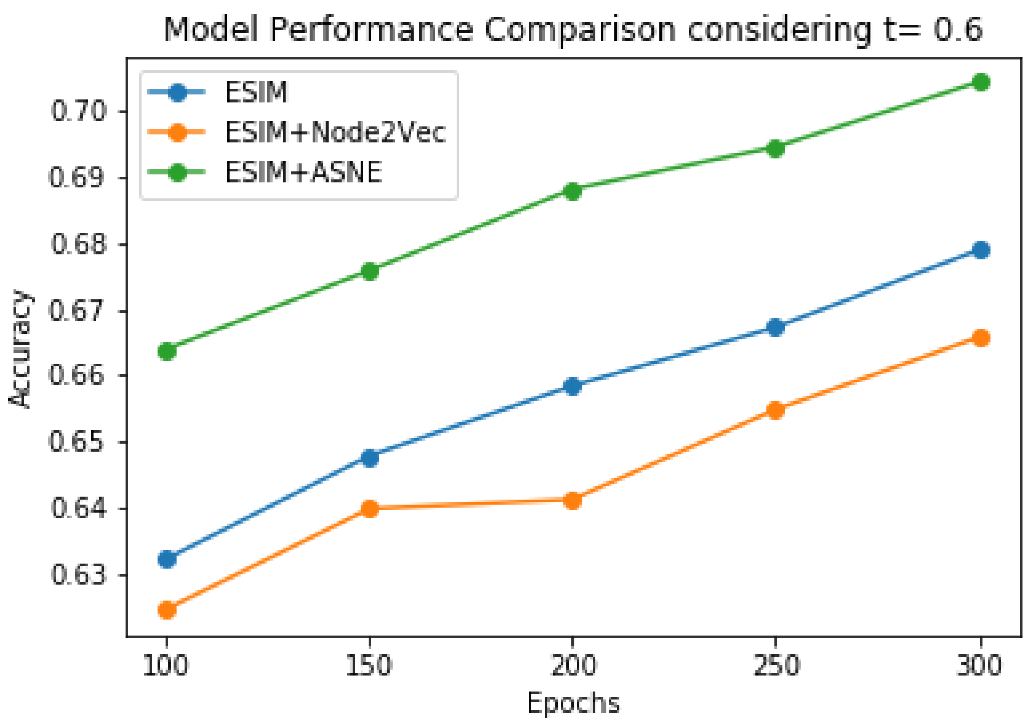

Figure 5.2: Model performance comparison.On the x-axis we used the number of epochs which is the number of training required in the model. On y-axis we used the accuracy of ESIM, and each of the enhanced ESIM models while considering the threshold value of 0.06 for each of them.

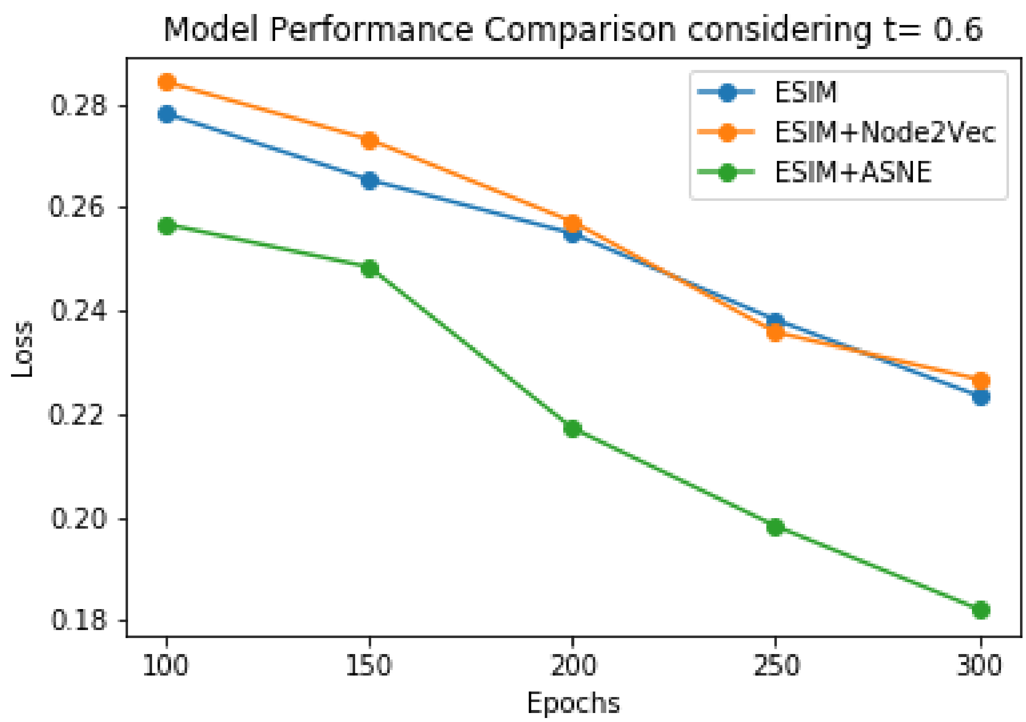

Figure 5.3: Model performance comparison.On the x-axis we used the number of epochs which is the number of training required in the model. On y-axis we used the log loss of ESIM, and each of the enhanced ESIM models while considering the threshold value of 0.06 for each of them.

CHAPTER 6: CONCLUSION

In this research, we present a new way to find a relationship between user interactions on social media which forms a dynamic network. We labeled the Gab interactions which had at least two users involved using NLTK Vader[1] compound score difference in interaction and varying threshold value t. We evaluated the labeling method by computing the net score and of each post in interaction and plotting the net score (net score of pos and neg score from NLTK Vader) of each post in an interaction against the net score of a post corresponding to the same interaction. By plotting net score we found a relationship that most of the interactions which were categorized as disagreement were accumulated near and below zero in the plot **??** while, the interactions which were categorized as agreement were accumulated near and above zero in the plot **??** implying that positive net score was correlated to the interactions labeled as agreement and negative otherwise. Further, we combine the text, and various network embeddings and innovatively integrated it in the state-of-art deep learning models. We measured the performance of each enhanced model with network features and compared the results. We evaluated the results by comparing the F1 score, accuracy, and recall of each enhanced model with the base model using the same dataset and its variation.

Future work includes improving the performance to predict the relationship between user interactions on social media as agreement or disagreement. The possible improvement in the model could be using transformers based deep learning model like BERT(Bidirectional Encoder Representations from Transformers)[52] instead of LSTM based models because recently it was proved to improve the performance of detecting sentiment from a text corpus. The possible improvement in labeling the dataset could be improved using transfer learning [53] which focuses on storing knowledge gained while solving one problem and applying it to a different but related problem. So the set of improvements like improving the labeling method and adapting to new deep learning frameworks like transformers and then solving this problem of predicting the relationship in interactions between users on social media in the form of agreement or disagreement.

REFERENCES

[1] C. J. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *Eighth international AAAI conference on weblogs and social media*, 2014.

[2] Q. Chen, X. Zhu, Z. Ling, S. Wei, H. Jiang, and D. Inkpen, "Enhanced lstm for natural language inference," *arXiv preprint arXiv:1609.06038*, 2016.

[3] C. Fellbaum, "Wordnet," *The encyclopedia of applied linguistics*, 2012.

[4] R. Rojas, "The backpropagation algorithm," in *Neural networks*, pp. 149–182, Springer, 1996.

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

[6] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *In EMNLP*, 2014.

[7] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th international conference on world wide web*, pp. 1067–1077, 2015.

[8] C. Yang, Z. Liu, D. Zhao, M. Sun, and E. Chang, "Network representation learning with rich text information," in *Twenty-Fourth International Joint Conference on Artificial Intelligence*, 2015.

[9] C. Tu, W. Zhang, Z. Liu, M. Sun, *et al.*, "Max-margin deepwalk: Discriminative learning of network representation.," in *IJCAI*, vol. 2016, pp. 3889–3895, 2016.

[10] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 701–710, 2014.

[11] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855–864, 2016.

[12] X. Sun, J. Guo, X. Ding, and T. Liu, "A general framework for content-enhanced network representation learning," *arXiv preprint arXiv:1610.02906*, 2016.

[13] L. Liao, X. He, H. Zhang, and T. Chua, "Attributed social network embedding," *CoRR*, vol. abs/1705.04969, 2017.

[14] H. Ito, T. Komamizu, T. Amagasa, and H. Kitagawa, "Network-word embedding for dynamic text attributed networks," in *2018 IEEE 12th International Conference on Semantic Computing (ICSC)*, pp. 334–339, IEEE, 2018.

[15] F. Harary, "On the notion of balance of a signed graph.," *Michigan Math. J.*, vol. 2, no. 2, pp. 143–146, 1953.

[16] D. Cartwright and F. Harary, "Structural balance: a generalization of heider's theory.," *Psychological review*, vol. 63, no. 5, p. 277, 1956.

[17] Y. Hou, J. Li, and Y. Pan, "On the laplacian eigenvalues of signed graphs," *Linear and Multilinear Algebra*, vol. 51, no. 1, pp. 21–30, 2003.

[18] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, "Propagation of trust and distrust," in *Proceedings of the 13th international conference on World Wide Web*, pp. 403–412, 2004.

[19] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Signed networks in social media," in *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1361–1370, 2010.

[20] N. Cesa-Bianchi, C. Gentile, F. Vitale, and G. Zappella, "A correlation clustering approach to link classification in signed networks," in *Annual Conference on Learning Theory*, vol. 23, pp. 34–1, Microtome, 2012.

[21] J. Tang, C. Aggarwal, and H. Liu, "Node classification in signed social networks," in *Proceedings of the 2016 SIAM international conference on data mining*, pp. 54–62, SIAM, 2016.

[22] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proceedings of the 19th international conference on World wide web*, pp. 641–650, 2010.

[23] P. Symeonidis, E. Tiakas, and Y. Manolopoulos, "Transitive node similarity for link prediction in social networks with positive and negative links," in *Proceedings of the fourth ACM conference on Recommender systems*, pp. 183–190, 2010.

[24] N. Ailon, M. Charikar, and A. Newman, "Aggregating inconsistent information: ranking and clustering," *Journal of the ACM (JACM)*, vol. 55, no. 5, pp. 1–27, 2008.

[25] P. Anchuri and M. Magdon-Ismail, "Communities and balance in signed networks: A spectral approach," in *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 235–242, IEEE, 2012.

[26] N. Bansal, A. Blum, and S. Chawla, "Correlation clustering. machine learning 56 (1-3), 89-113," *Google Scholar Google Scholar Digital Library Digital Library*, 2004.

[27] C. Swamy, "Correlation clustering: maximizing agreements via semidefinite programming.," in *SODA*, vol. 4, pp. 526–527, Citeseer, 2004.

[28] J. Tang, C. Aggarwal, and H. Liu, "Recommendations in signed social networks," in *Proceedings of the 25th International Conference on World Wide Web*, pp. 31–40, 2016.

[29] J. Tang, Y. Chang, C. Aggarwal, and H. Liu, "A survey of signed network mining in social media," *ACM Computing Surveys (CSUR)*, vol. 49, no. 3, pp. 1–37, 2016.

[30] D. Lo, D. Surian, K. Zhang, and E.-P. Lim, "Mining direct antagonistic communities in explicit trust networks," in *Proceedings of the 20th ACM international conference on Information and knowledge management*, pp. 1013–1018, 2011.

[31] M. Gao, E.-P. Lim, D. Lo, and P. K. Prasetyo, "On detecting maximal quasi antagonistic communities in signed graphs," *Data mining and knowledge discovery*, vol. 30, no. 1, pp. 99–146, 2016.

[32] L. Chu, Z. Wang, J. Pei, J. Wang, Z. Zhao, and E. Chen, "Finding gangs in war from signed networks," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1505–1514, 2016.

[33] T. Derr, Y. Ma, and J. Tang, "Signed graph convolutional networks," in *2018 IEEE International Conference on Data Mining (ICDM)*, pp. 929–934, 2018.

[34] K. Darwish, W. Magdy, and T. Zanouda, "Improved stance prediction in a user similarity feature space," in *Proceedings of the 2017 IEEE/ACM international conference on advances in social networks analysis and mining 2017*, pp. 145–148, 2017.

[35] I. Himelboim, S. McCreery, and M. Smith, "Birds of a feather tweet together: Integrating network and content analyses to examine cross-ideology exposure on twitter," *Journal of computer-mediated communication*, vol. 18, no. 2, pp. 154–174, 2013.

[36] D. McKendrick and S. A. Webb, "Taking a political stance in social work," *Critical and Radical Social Work*, vol. 2, no. 3, pp. 357–369, 2014.

[37] I. Augenstein, A. Vlachos, and K. Bontcheva, "Usfd at semeval-2016 task 6: Any-target stance detection on twitter with autoencoders," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pp. 389–393, 2016.

[38] K. Dey, R. Shrivastava, and S. Kaushik, "Topical stance detection for twitter: A two-phase lstm model using attention," in *European Conference on Information Retrieval*, pp. 529–536, Springer, 2018.

[39] S. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "Semeval-2016 task 6: Detecting stance in tweets," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pp. 31–41, 2016.

[40] M. Mohtarami, R. Baly, J. Glass, P. Nakov, L. Màrquez, and A. Moschitti, "Automatic stance detection using end-to-end memory networks," *arXiv preprint arXiv:1804.07581*, 2018.

[41] R. Dong, Y. Sun, L. Wang, Y. Gu, and Y. Zhong, "Weakly-guided user stance prediction via joint modeling of content and social interaction," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 1249–1258, 2017.

[42] M. Dias and K. Becker, "Inf-ufrgs-opinion-mining at semeval-2016 task 6: Automatic generation of a training corpus for unsupervised identification of stance in tweets," in *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pp. 378–383, 2016.

[43] W. Wei, X. Zhang, X. Liu, W. Chen, and T. Wang, "pkudblab at semeval-2016 task 6: A specific convolutional neural network system for effective stance detection," in *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, pp. 384–388, 2016.

[44] A. Benton and M. Dredze, "Using author embeddings to improve tweet stance classification," in *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, pp. 184–194, 2018.

[45] K. Darwish, P. Stefanov, M. Aupetit, and P. Nakov, "Unsupervised user stance detection on twitter," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 14, pp. 141–152, 2020.

[46] M. Lai, D. I. H. Farías, V. Patti, and P. Rosso, "Friends and enemies of clinton and trump: using context for detecting stance in political tweets," in *Mexican International Conference on Artificial Intelligence*, pp. 155–168, Springer, 2016.

[47] W. Magdy, K. Darwish, N. Abokhodair, A. Rahimi, and T. Baldwin, "# isisisnotislam or# deportallmuslims? predicting unspoken views," in *Proceedings of the 8th ACM Conference on Web Science*, pp. 95–106, 2016.

[48] A. Bagavathi, P. Bashiri, S. Reid, M. Phillips, and S. Krishnan, "Examining untempered social media: Analyzing cascades of polarized conversations," in *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM â19, (New York, NY, USA), p. 625â632, Association for Computing Machinery, 2019.

[49] C. Li, X. Niu, A. Al-Doulat, and N. Park, "A computational approach to finding contradictions in user opinionated text," in *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pp. 351–356, 2018.

[50] J. Kobbe, "Debatepedia: Claims of the arguments paired to the title question," 2019.

[51] A. Grover and J. Leskovec, “Node2vec: Scalable feature learning for networks,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD â16, (New York, NY, USA), p. 855â864, Association for Computing Machinery, 2016.

[52] L. Tian, X. Zhang, Y. Wang, and H. Liu, “Early detection of rumours on twitter via stance transfer learning,” in *European Conference on Information Retrieval*, pp. 575–588, Springer, 2020.

[53] K.-H. Lee, X. He, L. Zhang, and L. Yang, “Cleannet: Transfer learning for scalable image classifier training with label noise,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5447–5456, 2018.