Sentiment Analysis of Tweets on Societal Topics

Thesis submitted in partial fulfilment of the requirements for the award of the degree of

Doctor of Philosophy

in

Computer Science and Engineering

by

Loitongbam Gyanendro Singh

Under the supervision of

Dr. Sanasam Ranbir Singh

Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
Guwahati - 781039 Assam India

November, 2021

Dedicated to

My beloved Parents,

and

My dear Brothers and Sisters

Acknowledgment

First and foremost, I would like to express my heartfelt gratitude to my supervisor, *Dr. Sanasam Ranbir Singh*, for his continuous encouragement, endless patience, and positive guidance during my doctorate research. His continuing support and guidance have inspired me to grow as a scholar and as a person. I will be forever grateful for the opportunity to work with him and always will be indebted to him.

I am grateful to my thesis doctoral committee members – *Dr. Ashish Anand*, *Dr. Amit Awekar*, and *Dr. Priyankoo Sarmah* – for their insightful remarks and suggestions that have helped to improve the quality and clarity of my work. I want to thank the heads of the Department of CSE during my Ph.D. journey at IITG - Prof. S.V. Rao and Prof. Jatin Deka - for providing me with facilities and resources, including conference travel assistance. I am grateful to Prof. SRM Prassana (IIT Dharwad) and Prof. Sukumar Nandi for their kind support in extending my research work in the funded project at IIT Guwahati. I am also grateful to the MeiTy, Government of India, for providing resources and funding for my research work. I am grateful to the Technical staff of the Department of Computer Science and Engineering - Mr. Nanu Alan Kachari, Mr. Bhriguraj Borah, Mr. Hemanta Kumar Nath, Mr. Raktajit Pathak, Mr. Pranjit Talukdar, and Mr. Nava Kumar Boro for their helpful assistance with any engineering-related concerns. I am grateful to Mrs. Gauri Khuttiya Deori, Mr. Monojit Bhattacharjee, and Mr. Prabin Bharali for efficiently managing administrative tasks. I am grateful to all the faculty members, staff, and security personnel for their constant assistance and support.

I am thankful to all my colleagues and friends during my journey as a Ph.D. scholar. I am indeed thankful to my fellow lab mates at the OSINT family - Ranjan, Neelakshi, Rajlakshmi, Hemanta, Jennil, Bornali, Anasua, Akash, Durgesh, Rajib Sir, Mala, Rahul, Deepen, Anurag, Amitabh, Roshan, and many more for creating a wonderful experience at my workplace. The stimulating discussions, brainstorming, and collectively working together significantly influenced my development as an independent researcher.

I am blessed to have good buddies - Lenin, Somorjit, Gishan, Jennil, Ranjan, Hemanta, Alakesh, Neelakshi, Arabindu, Naro, Moa, Nini, Ato, Alex, Pankaj, Kamal, Nayan, and Pawan with whom I have shared some indelible moments of my life at IITG. I have enjoyed gathering with the Manipuri family in IITG for all the events we have organized and participated in. I want to thank my undergraduate and school days friends - Ashin, Anil, Amarjit, Henry, James, Nirosh, Sanatomba, Ronald, Gupta, W Sanatomba, Henba, and Milan for the beautiful memories and

cherished moments we have had together.

Finally, but most importantly, I want to thank the Almighty God and my family - Mom, Dad, my younger brothers - Alex and Monish, and my little sisters - Zadie, Mangal, and Kristina - for their endless love, support, warmth, compassion, and encouragement throughout the years. I am truly indebted to them.

November 15, 2021 Loitongbam Gyanendro Singh

Declaration

I certify that

- The work contained in this thesis is original and has been done by myself and under the general supervision of my supervisor.
- The work reported herein has not been submitted to any other Institute for any degree or diploma.
- Whenever I have used materials (concepts, ideas, text, expressions, data, graphs, diagrams, theoretical analysis, results, etc.) from other sources, I have given due credit by citing them in the text of the thesis and giving their details in the references. Elaborate sentences used verbatim from published work have been clearly identified and quoted.
- I also affirm that no part of this thesis can be considered plagiarism to the best of my knowledge and understanding and take complete responsibility if any complaint arises.
- I am fully aware that my thesis supervisor is not in a position to check for any possible instance of plagiarism within this submitted work.

November 15, 2021 Loitongbam Gyanendro Singh

Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
Guwahati - 781039 Assam India

Dr. Sanasam Ranbir Singh
Associate Professor
Email : ranbir@iitg.ac.in
Phone : +91-361-2582369

Certificate

This is to certify that this thesis entitled "**Sentiment Analysis of Tweets on Societal Topics**" submitted by **Loitongbam Gyanendro Singh**, in partial fulfilment of the requirements for the award of the degree of Doctor of Philosophy, to the Indian Institute of Technology Guwahati, Assam, India, is a record of the bonafide research work carried out by him under my guidance and supervision at the Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Assam, India. To the best of my knowledge, no part of the work reported in this thesis has been presented for the award of any degree at any other institution.

Date: November 15, 2021
Place: Guwahati

Dr. Sanasam Ranbir Singh
(Thesis Supervisor)

Sentiment Analysis of Tweets on Societal Topics

Abstract

Sentiment analysis is the task of classifying orientation of opinion (towards positive, negative, or neutral) expressed in a given piece of text. With the increasing availability of public opinions on various social media platforms such as Twitter, Facebook, LinkedIn, Google Plus, YouTube, etc., a surge in attention of data scientists/agencies in understanding public opinions on various social issues is evident in recent time. Understanding public opinions on various social issues is vital for various communities like business associates, policymakers, law enforcement agencies, etc. Though earlier studies of sentiment analysis generally consider well-structured texts written in a controlled environment, recent studies on sentiment analysis tasks mostly focus on social media data. Unlike regular texts, sentiment analysis of social media texts (micro-blogs in particular) needs to deal with various challenges. Micro-blogs are generally short in nature and often under-specified due to character limits. They often contain noise due to the presence of informal writing (shorten/elongated text), misspelling, multilingual code-switch and code-mixed contents. Among several social media platforms, Twitter has become one of the most popular micro-blogging platform today, and many government, non-government and commercial organizations use it for various purposes such as public announcements, event organizations, opinion polls etc. Because of its growing popularity and ease of data usage policies, majority of the recent studies on sentiment analysis of micro-blogs consider tweets collected from Twitter.

Like any other user-generated micro-blogs, tweets are short, under-specified, noisy, and multi-lingual. Studies have adopted various approaches to deal with the above issues - text normalization to remove noises in the text, using sentiment oriented emojis/hashtags, user's historical sentiment orientation profiling, downstream task-oriented fine-tuning of tweet embedding, multi-modal (text, embedded image/video, network) approach of combining features. It is evident from earlier studies that hashtags provide useful meta information linking a tweet to its underlying themes or topics. In addition to text, some of the earlier studies have also exploited network characteristics of tweets for better representation learning. Motivated by such studies, the objective of this thesis is to study the effectiveness of exploiting *hashtags* and *network representation learning* for *sentiment analysis of tweets on societal topics*. A *societal* topic can be defined as an event/issue that influences the general population within a society and attracts

their views. Earlier studies on sentiment analysis mostly focus on commercial domains such as product review, movie review, restaurant review, etc. Sentiment analysis is a domain-dependent task. A sentiment classifier built for a domain may not be suitable for another domain because of differences in the characteristics of sentiment-bearing indicators such as vocabulary, text construction, aspects, and their relationship. Sentiment analysis of public opinions on societal domain faces unique challenges because of wide ranges of possible topics leading to diverse vocabularies and target aspects, diverse text and language constructs, high volume of sarcastic texts, etc.

Many of the earlier social media data analytic studies in the societal domain consider public sentiment an important feature. Almost all such studies consider publicly available off-the-shelf sentiment analysis tools such as Emolex, Vader, SentiWordnet, etc., to determine the sentiment orientation of the opinions. As most such tools are not developed for societal domains, the suitability of using such off-the-shelf tools is subject to investigation. Motivated by this, the thesis first analyzes the characteristics of the tweets in societal* and non-societal topics. It investigates the performance of publicly available off-the-shelf tools and different in-house machine learning methods to understand better the similarities and differences of the tweets between the societal and non-societal topics. It is observed that the nature of the tweets of public opinions in societal topics is different from that of non-societal topics, and most of the off-the-shelf tools and classifiers built on non-societal topics are not suitable for sentiment classification of tweets in the societal topics.

Hashtags in tweets are provided by the person who posted the tweets to connect the tweets with their underlying themes or topics. It is also observed from earlier studies that many of the hashtags inherently bear sentiment polarity. Motivated by this, the second contribution of the thesis proposes a multi-tasking based method called *Sentiment Hashtag Embedding* (SHE) to identify sentiment associated with the hashtags. From various experimental observations, it is observed that sentiment analysis of tweets in the societal domain can be significantly improved by incorporating sentiment associated with the hashtags compared with its counterparts. An interesting observation of this study is that the proposed method is language independent and is able to discover the sentiment polarity of the hashtag in different languages.

Since tweets are generally short, under-specified, noisy, and multi-lingual, filtering tweets by adding sentiment-oriented tokens and removing insignificant tokens will help improve sentiment classification performance. Motivated by this, the third contribution of the thesis proposes a multi-layer network representation of a tweet and a heterogeneous multi-layer network (a layered network of hashtags,

*Creation of the annotated datasets over societal topics is supported by MeiTy, Government of India

mentions, and keywords) embedding to exploit and capture relational characteristics. Network representation is easier to identify significant and insignificant nodes and perform expansion or removal. From various experimental setups, it is observed that the sentiment classification performance improves and is more immune to under-specificity, noise, and multi-linguality.

Though the sentiment classification performance improves with the above contributions, a systematic study of incorporating textual and structural features using various state-of-the-art embedding methods has not been investigated. The fourth contribution of the thesis proposes an end-to-end multi-views representation learning method to incorporate the textual and graphical representation of the tweets systematically. From various experimental results, it is evident that the proposed method significantly improves after incorporating both the textual and graphical views compared to single view representation.

Contents

Listing of figures

List of Tables

List of Abbreviations

Terms	Abbreviations
A-Boost	AdaBoost
AE	Autoencoder
API	Application Programming Interface
BERT	Bidirectional Encoder Representation from Transformer
Bi-LSTM	Bidirectional LSTM
BOW	Bag of words
CB	Continuous BOW
CNN	Convolution Neural Network
CNN-BiLSTM	CNN + Bi-LSTM
DGCNN	Deep Graph Convolution Neural Network
DNN	Deep neural network
DT	Decision Tree
DW	DeepWalk
ET	Extra Trees
FT	FastText
GB	Gradient Boosting
GCN	Graph Convolution Network
HV	Hashtag2Vec
IBM	International Business Machines
kNN	k-Nearest Neighbour
LR	Logistic Regression
LSA	Latent Semantic Analysis
LSTM	Long Short Term Memory
MLP	Multi Layer Perceptron
MNE	Multiplex Network Embedding
MVE	Multi-View Embedding
N2V	Node2Vec
PMI	Pointwise Mutual Information
POS	Part-of-Speech
QoL	Quality of Life

RF	Random Forest
RNN	Recurrent Neural Network
SA	Sentiment Analysis
Seg-BERT	Segmented Graph-BERT
SG	SkipGram
SHE	Sentiment Hashtag Embedding
SNE	Sentiment polarized Node Expansion
SOA	Strength of word Association
SO	Semantic Orientation
SVM	Support Vector Machine
VR	Verse

The secret of getting ahead is getting started.

Mark Twain, American writer

1 Introduction

Sentiment analysis is a natural language processing task that focuses on determining the sentiment orientation (positive, negative, or neutral) of a person's view on a targeted topic or entity at a particular time. An opinion holder can convey their thoughts on any issue or entity and its various aspects. The opinion conveyed can be in the form of text, speech, or video. This thesis work takes into account textual opinion to perform the Sentiment Analysis (SA) study. There are several levels of textual granularity where SA can be applied, such as document, sentence,

and aspect. Document-level SA research seeks to detect the overall sentiment of the text document; sentence-level research attempts to identify the sentiment of each sentence. In contrast, the aspect-level SA analysis detects the sentiment conveyed in the aspects of an entity or a subject. For example, in an opinion sentence ***O1***: "*The iPhone's call quality is good, but its battery life is short*", there are two aspects of *iPhone* i.e., '*call quality*' and '*battery life*'. At the aspect level, the '*call quality*' has positive sentiment. In contrast, the '*battery life*' has negative sentiments. At the sentence level, it can be a neutral sentiment given equal weightage to each aspect. Contrarily, the sentence-level sentiment can be biased by how much weightage is given to each component. For example, emphasizing the sentiment of the *iPhone's* '*call quality*' can alter the sentence-level sentiment and vice versa.

Research in sentiment analysis can be dated back to 1976 when the notion was first proposed[17,134]. With the rise of Web 2.0* and the proliferation of social media platforms, the popularity of this research area continued to expand in the early 2000s[95,123]. In the last two decades, a plethora of researchers has contributed to the field of sentiment analysis. However, with the advent of various social media platforms, new challenges continue to proliferate. With user-generated content (public opinions in general) becoming more readily accessible on social media platforms, sentiment analysis of public opinions has become increasingly important for many agencies such as data analysts, social scientists, business corporates, government departments, and so on. Depending on the type of platform, the user-generated content on the social media platforms also varies. Social net-

*https://en.wikipedia.org/wiki/Web_2.0

working sites (e.g., Facebook*, LinkedIn†), image sharing sites (e.g., Instagram‡, Pinterest§), microblogging sites (e.g., Twitter¶, Tumblr‖), video sharing sites (e.g., Youtube**, TikTok††, Likee‡‡), and discussion forum (e.g., Quora§§, Reddit¶¶) are some of the instances of different type of social media platforms. Among the various social media platforms, Twitter, a microblogging service, has emerged as one of the possible sources of information for academic researchers, policymakers, politicians, celebrities, and the general public. Because of its growing importance and ease of data usage policies***, recent sentiment analysis studies on social media data consider tweets collections as potential experimental datasets. Micro-blogs are often brief in length, and individuals tend to express their opinions without devoting much time to reading and creating blogs†††. Since the micro-blog text is generally short, recent sentiment analysis studies on micro-blog data classify the sentiment at the blog level and its underlying aspects. This thesis work focuses on classifying the sentiment of tweets at the blog level.

1.1 Background

According to Bing Liu[63], an opinion can be defined as a quintuple (e_i, a_{ij}, s_{ijkl}, h_k, t_l) where e_i is the name of an entity, a_{ij} represents an aspect of e_i, s_{ijkl} is the

*https://www.facebook.com/
†https://in.linkedin.com/
‡https://www.instagram.com/
§https://in.pinterest.com/
¶https://www.twitter.com/
‖https://www.tumblr.com/
**https://www.youtube.com/
††https://www.tiktok.com/
‡‡https://likee.video/
§§https://www.quora.com/
¶¶https://www.reddit.com/
***https://developer.twitter.com/en/developer-terms/policy
†††https://blog.twitter.com/en_us/topics/product/2017/tweetingmadeeasier.html

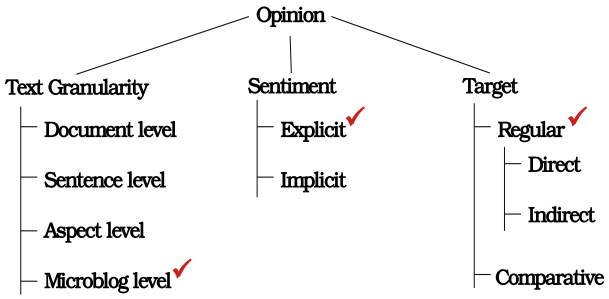


Figure 1.1: Type of opinions from text granularity, sentiment, and target perspective. Tick marks indicates the type of opinions considered in the scope of the thesis.

sentiment expressed towards an aspect a_{ij} of the entity e_i at time t_l by an opinion holder h_k. The sentiment polarity of an opinion is generally measured as positive, negative, or neutral. There are several types of opinions with respect to the type of textual granularity, sentiment, and target. Figure 1.1 shows different type of opinions from various perspective. From a sentiment perspective, an opinion can be categorized into *explicit* and *implicit* type of opinions. Explicit opinion clearly expresses the sentiment polarity of the opinion holder using sentiment indicating terms. For example, an opinion about `abortion`: "*It is not murder I don't believe because a fetus IS NOT a baby.*"[46]. Here the opinion is an explicit opinion where the opinion holder is explicitly favouring abortion by mentioning the sentiment indicating terms like *murder*, *believe*, etc. There is no specific sentiment indicating terms or references to the target in implicit opinion, but the sentiment polarity of the opinion holder can be understood implicitly. For example, an opinion "*the life of the fetus is not important.*" is an implicit opinion where the opinion holder favors abortion implicitly without expressing any sentiment indicating terms. This study takes into consideration both *explicit* and *implicit* opinions.

From a target perspective, opinion could be classified into two types, namely *regular opinion* and *comparative opinion*. The *regular opinion* is a simple opinion

where the opinion holder expresses his views on an entity or its aspects. It can be further divided into two sub-types: *direct opinion* and *indirect opinion*. As the names suggest, *direct opinion* and *indirect opinion* are expressed directly and indirectly on an entity or its aspects. For example, let us consider two tweets ***O2***: "*Use Signal*" and ***O3***: "*Bought some Dogecoin for lil X, so he can be a toddler hodler*" posted by Elon Musk*. Here ***O2*** is a direct opinion where the opinion holder (Elon) asks his follower to use the *Signal*† messaging application. Whereas, ***O3*** is an indirect opinion where he is announcing his follower about the investment to *Dogecoin*‡, a cryptocurrency company, making an indirect suggestion to his followers to invest in the same company. The *comparative opinion* is a type of opinion that an opinion holder expresses by comparing two or more entities or their aspects, considering their similarities or differences. For example, in tweet ***O4***: "*BJP won the no-confidence motion. @RahulGandhi won the confidence. #RahulHugsPM #NoCofidenceMotion #BhukampAaGaya #HugDiplomacy #HugDay*", the opinion holder has compared the two entities *BJP*§ and *@RahulGandhi*¶ where the opinion holder expresses negative sentiment towards *BJP* while positive sentiment towards *@RahulGandhi*. Classifying the sentiment for the comparative type of opinions is more focused on aspect-based sentiment analysis. This study attempts to classify the sentiment of tweets at the blog level with *regular* type of opinion.

Most of the earlier studies of sentiment analysis focus on the commercial domains like product reviews[149,36,26], movie reviews[69,99], hotel reviews[111], etc. How-

*CEO of SpaceX, Tesla Companies

†https://signal.org

‡https://dogechain.info

§One of the major political parties of India.

¶President of the Indian National Congress political party.

ever, as the number of active Twitter users increases in sharing opinions on various societal topics such as social issues, political views, government policies, social unrest, and so on, analyzing public sentiment toward target topics has become increasingly important for government, non-government, and commercial entities dealing with social, political, and economic issues. Twitter has become one of the potential platforms for public announcements, event organizations, opinion polls, etc. For example, tweets by Elon Musk: ***O2***: "*Use Signal*", ***O3***: "*Bought some Dogecoin for lil X, so he can be a toddler hodler*" can change the fortunes of companies like Signal and Dogecoin. Besides, spreading negativity on Twitter can also bring chaos to society. For example, Twitter has banned U.S. President Trump from using their services due to riots by his supporters in U.S. Capitol following his tweet ***O5***: "*Statistically impossible to have lost the 2020 Election. Big protest in D.C. on January 6th. Be there, will be wild!*"*. In such cases, understanding public opinions on the societal topic discussion on Twitter is essential for various communities like businesses, policymakers, law enforcement agencies to account for decision making, reshaping businesses, sway political issues, etc.

A *societal topic* can be defined as a topic that influences many of the general population in a society. As observed from the study of Karamibekr and Ghorbani[46], social issues are usually related to other sub-issues or topics while products usually have defined features. The aspects of the products are well defined and are often explicitly mentioned in the opinions making the extraction of relevant sentiment indicative features easier. In comparison, societal topics and their aspects are not predefined but evolve with time. It makes the handling of

*https://www.usatoday.com/story/news/nation/2021/01/04/january-6-dc-protests-against-election-certification-could-v
4132441001/

tweets in societal topics more challenging. For example, given two opinions, ***O1***: *"The iPhone's call quality is good, but its battery life is short"* and ***O6***: *"@timesnow first we should clean up our home in which such burhan supporters are still living.. sweep them"*, we can easily identify the features of the products *iPhone* from ***O1***, i.e. *call quality* and *battery life*. Whereas in the second opinion ***O6***, it is not straightforward to identify the features. Therefore, sentiment analysis of public opinions on the societal domain faces unique challenges because of wide range of possible topics such as political, policies, social unrest, climate change, etc.

Sentiment analysis is a highly domain-dependent task i.e., a sentimental word in a domain can be opposite sentiment in another domain[94,46,62,63,35,104]. For example, consider an opinion from movie review domain ***O7***: '*The plot of Dunkirk movie is heavy #awesome*' and an opinion from product review domain ***O8***: '*Motorola Onepower is really heavy #sucks*'. The word *heavy* is being used to express the sentiment of both opinions ***O7*** & ***O8*** but of different sentiments i.e. positive and negative. The opinion of the movie review domain ***O7*** is of positive sentiment while the product review opinion ***O8*** is of negative sentiment. Therefore, a classifier built for product review domain may not be effectively used in movie review domain. Hence, for building an effective sentiment classifier of a particular domain requires a considerable amount of sentiment annotated corpus of the domain.

Building sentiment analysis classifiers have gone through various paradigm shifts, from statistical methods[123,124] to rule-based[99], lexicon-based[118,9,78], feature-based[95,53,10], and deep neural network[48,109] approaches. The statistical approaches identify the association between words and sentiment-annotated documents using

Point-wise Mutual Information* or Latent Semantic Analysis† techniques. The rule-based methods infer the sentiment of an input sentence or document by using a sentiment lexicon and part of speech information of the input sentence or document. As an alternative to utilizing a part of speech tagger, lexicon-based methods infer the sentiments expressed in a document using a readily available sentiment lexicon. The above approaches can be used with little or no training corpus if a sentiment lexicon and part-of-speech tagger are provided. However, because of the context-sensitivity of human language, such approaches fall short of covering all the rules requiring expert knowledge. A machine learning-based method, on the other hand, would be suited to overcome these obstacles. Machine learning (especially supervised learning) can adapt and build a learning model for specific purposes and circumstances depending on the domain of the training data. As a result, traditional machine learning-based classifiers rely on features to discriminate between sample classes. In order to work effectively, such classifiers require proper feature engineering strategies[106,93]. Recent studies on SA have used various deep learning frameworks to eliminate feature engineering challenges[88,44,91,109,29]. They have shown comparable or better performance than traditional methods such as SVM, Logistic Regression (LR), and Random Forest (RF). Therefore, most of the recent sentiment analysis studies focus on neural network models[35].

*https://en.wikipedia.org/wiki/Pointwise_mutual_information
†https://en.wikipedia.org/wiki/Latent_semantic_analysis

1.2 Challenges

Since Twitterers can tweet using any language of their choice, sentiment analysis of tweets has to deal with a wide range of challenges. For example, the challenges include under-specificity due to short informal text (tweets might be too short and hard to comprehend without context information), noise due to informal writing (no specific rules of writing), misspelling, and multilingual code-mixed and code-switch contents. Researchers have attempted to address the above challenges using various approaches such as task-specific representation learning[113,97,30,119,48], incorporating additional information such as hashtags[6,101], user relationships[148], multi-source information[149,69], ensembling[5,8,130], etc. The task-specific representation learning methods attempt to encode sentiment information into the semantic representation of words to enhance the sentiment classification task. Earlier studies have shown that incorporating additional information such as hashtags[6,101,131], user relations[148], etc., can further enhance feature representation for sentiment classification tasks. To incorporate such information, various studies have investigated techniques like multi-view learning[149,69], ensembling[5,8,130], multi-task learning[105,66,45], transfer learning (pre-trained embedding learned from the text used in network representation learning and vice-versa)[67] for sentiment classification.

Various studies have explored the importance of using hashtags in social media data[71,6,54,101,131]. A hashtag is a particular form of keyword or phrase that begins with '#', acting as meta-data of users' tweets to reflect the users' views. Studies have shown that hashtags help in linking tweets with its underlying theme or topic. Wang et al.[131] have categorized the usage of hashtags into three different categories: i) topic hashtag – to represent topics such as *#UriAttack*, *#GSTN*,

ii) sentiment hashtag – to represent sentiment or emotion such as *#happy*, *#sad*, and iii) topic-sentiment hashtag – to represent topic with emotion such as *#RahulHugsPM*, *#FarmersProtest*. It is observed that tweets with similar hashtags can have different sentiments. For example, ***O4***: "*BJP won the no-confidence motion. @RahulGandhi won the confidence. #RahulHugsPM #NoCofidenceMotion #BhukampAaGaya #HugDiplomacy #HugDay*" and ***O9***: "*BJP's attempt to bring about change in our society be it in our education, maturity, behaviour, thinking pattern gets a blow when its senior leaders cannot keep a restrain on their behaviour #NoConfidenceMotion #RahulGandhi #IndiaTrustsModi #RahulHugsPM #NoConfidencePolitics*" share similar hashtags *#RahulHugsPM* and *#NoConfidenceMotion*. However, the sentiment of ***O4*** has negative polarity while ***O9*** has positive polarity towards *BJP*[*]. It is further observed that ***O4*** and ***O9*** have sentiment indicating hashtags like *#NoConfidenceMotion*, *#IndiaTrustModi*, *#NoConfidencePolitics*. In such cases, it is beneficial to identity the sentiment of the semantic relation of hashtags which can aid in classifying the sentiment of tweets.

When a tweet is (very) short, it is not easy to understand the underlying opinion of the tweet without providing contextual information. For example, Elon Musk[†] posted a tweet ***O2***: *"Use Signal"*. This tweet is under-specified as it requires the reader to understand what *Signal* he is referring to. The author may have posted this tweet to prompt his follower to use `Signal messaging app`[‡]. Under-specificity in this tweet may have resulted in benefiting the company named `Signal Advance`[§] due to the confusion in the contextual information. In such a

[*]One of the major political parties of India
[†]CEO of SpaceX, Tesla Companies
[‡]https://signal.org/
[§]https://www.bloomberg.com/news/articles/2021-01-11/musk-sowed-ticker-confusion-sends-medical-device-maker-up-5-100

case, adding contextual information to the tweet can enrich the understanding of the tweet. For example, in the following tweet ***O10***: *"@elonmusk was right again get signal #deletefacebook"*, it is clear that the word *signal* is related to social media and the author have negative sentiment towards `Facebook`. This made the reader of ***O2*** clear that *@elonmusk* has positive sentiment towards `Signal messaging app`. However, adding or filtering tokens in a text sequence is not a straightforward task. Using a network-based representation of the tweet can make adding or removing nodes from the network more accessible. As *#hashtags* and *@mentions* connect tweets surrounding the same topic or theme, linking hashtags, mentions, and keywords can enhance network-based tweet representation.

1.3 Research Objective

Motivated by above observations, the objective of this thesis is to study the effectiveness of exploiting hashtags and network representation learning for *sentiment analysis of tweets on societal topics*. There is a lack of studies on building sentiment classifiers for tweets on societal topics. Most of the earlier studies of tweet analysis on societal topics mainly focus on pre-defined topics such as *Climate Change, Election*, and entities involved for quantifying the sentiments and time series analysis[16,87,116,92,59,126,7]. Almost all of such studies consider publicly available off-the-shelf sentiment analysis tools. To name a few, studies[55,87,59,73] use SentiStrength* to analyze public sentiment on the issues like political analysis, natural disaster, societal domains, etc. MeaningCloud† was used in Singh et al.[116] for analyzing public opinion related to government policies. Öztürk and Ayvaz[92]

*http://sentistrength.wlv.ac.uk
†https://www.meaningcloud.com

use RSentiment* to analyze public sentiment on social unrest issues. It is observed from the studies of Maynard et al.[73] and Ribeiro et al.[104] that these tools are developed based on heuristic assumptions such as sentiment lexicon — sentiment lexicon is specific to a particular domain as words can represent different meaning in different domains[85]. Hence, the performances of the tools may differ from one domain to another domain. Therefore, this thesis aims to address the challenges of building an effective sentiment analysis classifier for tweets on the societal domain and find possible solutions by incorporating hashtags and relational structures. More specifically, the thesis attempts to address the following objectives:

- As there is a wide range of issues in the societal domain, the first objective is to understand the nature of the tweets (such as usage of tokens like hashtags, mentions, keywords) concerning societal and non-societal topics and across different issues in societal topics. Further, investigate the need to build an effective sentiment classifier for the societal domain by exploring the performance of the existing off-the-shelf SA tools with the inhouse build classifiers.

- While posting opinions on Twitter, users often use hashtags to reflect meta-information such as sentiment, emotion, topic, and entity, etc. Understanding hashtags help in addressing various issues related to opinion and text mining tasks such as topic modeling[131,71], sentiment classification[6], sentiment lexicon generation[54,101,80], stance detection[140,81], etc. To enhance the performance of sentiment analysis task, it is desirable to incorporate senti-

*https://cran.r-project.org/web/packages/RSentiment/index.html

ment information over the semantic representation of the hashtags. Therefore, the study's second objective is to encode sentiment-specific embedding of hashtags to enhance the performance of sentiment classification tasks while preserving its semantic information.

- Sentiment analysis of tweets usually suffers from the problem of under-specificity, noise, and multi-lingual content. To address the above challenges, earlier studies have attempted to incorporate additional information into the tweet. However, adding additional information in the text sequence is not straightforward. Therefore, the third objective of the study is to mechanize a method to incorporate additional information such as through network perspective by exploiting hashtag, mention, and keyword relations.

1.4 Contributions

This thesis work aims to address the challenges of sentiment analysis of tweets on societal topics by exploiting hashtags and network representation learning approaches. This thesis has made four contributions.

- The thesis first analyzes the characteristics of the tweets in societal and non-societal topics, and investigates the performance of publicly available off-the-shelf tools and different in-house machine learning methods over different datasets from societal and non-societal topics.
- The second contribution of the thesis proposes a multi-tasking based method called Sentiment Hashtag Embedding (SHE) to encode sentiment information while preserving the semantic characteristics to enhance the sentiment classification task.

- The third contribution of the thesis proposes a sentiment classification method using heterogeneous multi-layer network representation of tweets incorporating the relations of hashtags, mentions, and normal keywords to address the problem under-specificity, noise, and multi-lingual challenges by adding sentiment polarized nodes and removal of non-polarized nodes in the heterogeneous network.

- Finally, the fourth contribution of the thesis proposes a framework for tweet sentiment classification task by incorporating both text and graph views through multi-view representation learning method.

1.5 Organization of the Thesis

The thesis has eight chapters. The thesis is organized as the following chapters.

- **Chapter 1 Introduction**: This chapter introduces the problem of sentiment analysis of tweets on Societal topics, the challenges involved, and the motivation of the thesis work. The research objective of this thesis work is formally discussed, followed by an overview of contributions made.

- **Chapter 2 Literature Review**: This chapter discusses the different sentiment analysis approaches such as the different approaches to perform sentiment analysis, prior studies on sentiment analysis of societal topics, handling under-specificity, multilingual, and noisy tweets.

- **Chapter 3 Characteristics of societal and non-societal datasets**: This chapter performs statistical analysis of the societal and non-societal datasets to understand the characteristics of word usages across the domains.

- **Chapter 4 Empirical Study of Sentiment Analysis Tools and Techniques on Societal Topics**: This chapter evaluates the performance of available off-the-shelf sentiment analysis tools and machine learning techniques over societal and non-societal datasets.

- **Chapter 5 Sentiment Hashtag Embedding Through Multitask learning**: In this chapter the second contribution of the thesis work is presented i.e., the proposed method of sentiment hashtag embedding through multi-task learning to encode sentiment information of hashtags while preserving its semantic characteristics.

- **Chapter 6 Sentiment Analysis of Tweets using Heterogeneous Multi-layer Network Representation**: In this chapter, the third contribution of the thesis work is presented, i.e., the proposed method that transform tweets into heterogeneous multi-layer networks to address the challenges of sentiment analysis of tweets such as under-specificity, noise, and multilingual content through node expansion and shrinking.

- **Chapter 7 Sentiment Analysis of Tweets using text and graph multi-views**: This chapter discusses the forth contribution of the thesis work, i.e., the proposed multi-view learning framework to incorporate different views of tweet for sentiment classification task.

- **Chapter 8 Conclusion and Future Work**: This chapter concludes with possible future research directions of this thesis.

A man's feet should be planted in his country, but his eyes should survey the world.

George Santayana, American philosopher

2 Literature Survey

Sentiment analysis is considered as one of the subfields of text mining where majority of the opinions are available in textual format[12]. The sentiment analysis study has been carried out in various perspectives, such as the text content, methodologies, and applications. Figure 2.1 shows an overview of the sentiment analysis studies. Studies on text content focus on the type of text format, i.e., document level, sentence level, aspect level, or tweet level. The methodological studies attempt to address the challenges of building a sentiment analysis model.

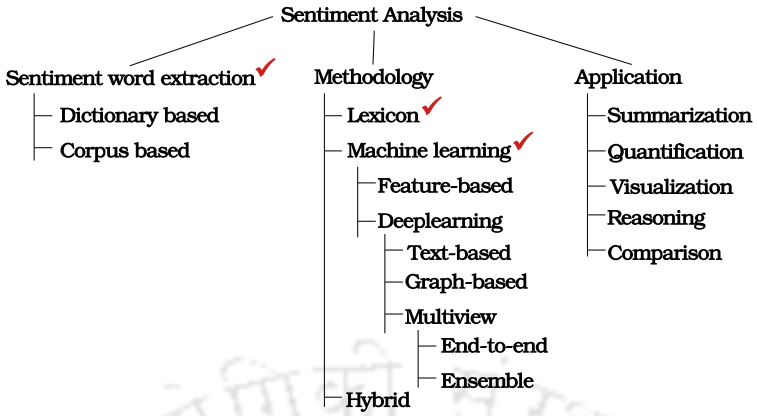


Figure 2.1: Type of sentiment analysis studies perform with respect to sentiment feature extraction, classification methods, and application. Tick marks indicates the type of studies considered in the scope of the thesis.

In comparison, the application-oriented studies focus on summarizing, visualizing, quantifying public sentiments over target topics. The primary objective of a sentiment analysis study is to identify the sentiment of a given opinionated text. Therefore, it requires identifying sentiment indicating tokens from the opinionated text to classify its sentiment. Then, the sentiment indicating tokens are exploited as features for training feature-based machine learning models or as a heuristic model to classify the sentiment of the opinionated text. Further, these classification models are used for sentiment summarization, comparison, quantification, and visualization purposes. In this chapter, we briefly discuss studies related to Twitter sentiment analysis, primarily focusing on the following three areas: (i) sentiment feature extraction, (ii) feature-based sentiment analysis techniques, and (iii) application of sentiment analysis methods on social issues.

2.1 Sentiment feature extraction

Sentiment polarize words are considered important features for the sentiment classification task. There are two commonly used approaches to extract sentiment

words or generate sentiment lexicon for a particular domain or language namely *Corpus-based* and *Dictionary-based* approaches[94,62]. *Corpus-based* approach requires linguistic knowledge or conventions on connectives patterns (e.g., and, or, but, etc.) to identify sentiment words and their orientations from the corpus. In contrast, the *Dictionary-based* approach uses some existing sentiment lexicon as seed words to populate the existing sentiment lexicon.

2.1.1 Corpus-based approaches

Several studies consider word-to-word similarity/distance as the basic measure for generating semantic lexicon using corpus-based approaches. Some of the approaches are Label Propagation, Random Walk on synonym and antonym network of words (such as Wordnet)[150,43], point-wise mutual information between two words[123], template matching in n-gram word sequence[125]. Further in several studies, authors attempt to capture semantic relation between words by projecting word representation into low dimensional latent space. Turney and Littman[124] uses Latent Semantic Analysis (LSA), while Maas et al.[72] uses Latent Dirichlet Allocation for estimating word-to-word association strength in low dimensional latent space.

The popularity of semantic word embedding methods such as Word2Vec[77] and C&W[25] have inspired to perform sentiment feature extraction by exploiting the semantic embeddings. Several studies have generated sentiment word embedding via semantic embedding following a two-tier approach, i.e. (i) generate semantic embedding using state-of-the-art embedding methods and (ii) encode sentiment polarity to the semantic embedding using supervised sentiment classification model[72,48,119,139,31]. To incorporate sentiment information, study in Kim[48]

uses a CNN based classifier over the above-mentioned pre-trained embeddings. In a similar direction, Tang et al.[119] exploit distant supervision using emoticons for encoding the sentiment polarity. Further, studies in Ye et al.[139] and Fu et al.[31] exploit the available sentiment lexicons (e.g. SentiWordNet) as supervised information for generating sentiment embedding.

2.1.2 Dictionary-based approaches

Studies[94,62] have used Label Propagation approach to populate sentiment lexicon. In this approach, two seed lists exist, one with positive polarity and the other with negative polarity. Each of the selected seed sentiment words is inspected through an existing dictionary such as WordNet* to find their synonym and antonym words. Each synonym word of the selected seed word is added to the respective sentiment category or seed list. The antonym words of the selected seed word, on the other hand, are placed to the opposite sentiment category or seed list. For example, a positive sentiment query seed word "*good*" has synonyms "*fine*, *virtue*" and antonyms "*bad*, *wicked*". The positive seed list is expanded by adding the synonym words "*fine*" and "*virtue*", while the negative seed list is expanded by adding the antonym words "*bad*" and "*wicked*". The new antonym and synonym words added to the respective categories are further used to generate synonym and antonym words. This iteration ends when no more new words from the corpus are added to the seed list.

Mohammad et al.[79] propose a computationally inexpensive approach to generate a high-coverage semantic oriented lexicon by making use of Roget-like thesaurus and a handful of antonym-generating affix patterns. Their approach does

*https://wordnet.princeton.edu/

not require any text corpora or manually annotated semantic-oriented labels.

Goyal and Daumé III[39] propose an approach to construct semantic orientation lexicons using a large corpus and a Roget-like thesaurus. In this approach, it finds semantic orientation (SO)[124] of a word and also used a Roget-like thesaurus structure in which near-synonymous words appear in a single group. This approach calculates the SO of each group with respect to the whole word and assigns the SO score of a group to individual words in the group.

Yazidi et al.[138] propose a novel algorithm for lexicon generation that supports better transitive sentiment polarity transferring from seed word to target words using the theory of Structural Balance Theory. The principles underlying structural balance are based on theories in social psychology*. The intuition of the algorithm is motivated by the concept *the enemy of my enemy is my friend* that preserves the transitivity structure captured by antonyms and synonyms. Their approach uses three thesauri as the source of information for lexicon generation.

2.2 Feature-based sentiment analysis

As previously stated, the primary goal of sentiment analysis is to classify the sentiment of a given opinionated text. Text classification using feature-based machine learning algorithms requires distinguishing features of a particular domain to classify the input samples. This section presents the literature on feature engineering for sentiment analysis on societal topics. The importance of feature engineering for sentiment analysis on societal issues has been reported in several studies[81,128,117,46]. Karamiberkr and Ghorbani[46] have investigated the usage of word vocabulary in product reviews and societal issues comments. They discovered that product re-

*https://en.wikipedia.org/wiki/Balance_theory

views have lesser text dynamics than societal problems discussions. The features for products and services are easily recognizable compared to societal issues in terms of specificity, mentions, and usage of part-of-speeches. Therefore, finding appropriate sentiment classification features on public opinion in the societal domain is a challenging task. Studies in Stance Detection* on societal issues have also investigated the effectiveness of using a feature-based classifier trained with various sentiment features such as sentiment lexicons, Part-of-Speech (POS), etc[81,117]. These studies have suggested the necessity for a suitable feature extractor to enhance the performance of SA classifiers on societal topics. Kouloumpis et al.[53] have explored hashtag-based corpus generation using Twitter-specific features. They demonstrated the effectiveness of using emoticons and Twitter-specific features in addition to n-gram features by evaluating the performance of SA classifiers (AdaBoost and SVM).

2.2.1 Use of deep learning methodologies

One of the challenges in building effective sentiment analysis tools using feature-based classifiers is the need to select appropriate features[106,93]. To get rid of feature engineering problems, recent studies on SA have exploited various deep learning frameworks and observed comparable or better performance as compared to traditional approaches like SVM, Logistic Regression (LR), and Random Forest (RF)[88,44,91,109,29]. Goldberg[37] and Zhang et al.[144] have discussed how deep learning approaches are used for natural language processing and SA. Here we present some of the literature that has motivated our study.

Various studies have used the combination of word embedding techniques such

*https://en.wikipedia.org/wiki/Stance_(linguistics)

as Word2Vec* and GloVe† and deep CNN to enhance the performance of SA classifiers[88,44,98,109,91]. They evaluated the performance of their proposed approaches with various classifiers such as SVM, LR, NB and RF etc. It is observed that the proposed approaches, which is a combination of word embedding techniques and CNN, have outperformed other classifiers on different datasets which were used in various SA studies[35,104].

Al-Smadi et al.[4] and Akhtar et al.[3] studies have compared the performance of traditional classifiers with various DNN classifiers. Al-Smadi et al.[4] have evaluated that SVM classifier outperformed RNN classifier in aspect-based sentiment classification of Arabic hotel's review datasets. Akhtar et al.[3] have evaluated that the hybrid of CNN and SVM classifiers outperformed individual classifier in sentiment classification of sentiment on four Hindi datasets of different domains. However individually, SVM classifier have outperformed CNN classifier. We also have similar observation as to these studies.

2.3 Application of sentiment analysis on societal topics

Cao et al.[16] and Lerman et al.[59] study the spatio-temporal sentiment pattern in the regions of USA. Cao et al.[16] study the quality of life (QoL) influence by land use and time period on public sentiment in Massachusetts, USA from 31 November 2012 to 3 June 2013. The IBM Watson Alchemy API was used to quantify the sentiment of people in the area. They observed different characteristics of the users' sentiment across different land use and time. The users' have higher sentiment in the commercial and public areas, during the noon/evening and on

*https://code.google.com/archive/p/word2vec/
†https://github.com/stanfordnlp/GloVe

the weekend. In contrast, users were more likely to show negative sentiment within the areas of farmland, transportation, and industry, around midnight and on weekdays. Lerman et al.[59] have studied how online social interactions are affected by psychological and demographic factors. They collect 4 months tweets from the Los Angeles, USA. To quantify the sentiment of the people opinions SentiStrength tool were used. They found that social media users who engaged more deeply with less diverse social contacts express more negative emotions to seek support while diverse social contacts share positive emotions.

Singh et al.[116] and Neppalli et al.[87] study the spatio-temporal sentiment distribution of people discussed in Twitter on the human-induced and natural disaster events. Singh et al.[116] have studied SA on the issue, demonetization of 500 and 1000 Indian currency notes, which is one of the social issue happened in India. They have used MeaningCloud API in their study to quantify the sentiment of the people. Neppalli et al.[87] have studied SA during the disastrous event Hurricane Sandy through tweets posted on Twitter. They used two binary classifiers for classifying neutral, positive and negative sentiments. For classifying neutral or subjectivity, they use SentiStrength method and for polarity classification they have used SVM classifier.

Öztürk and Ayvaz[92] and Garg et al.[33] study SA on social unrest events of Syria crisis and Uri attack. Öztürk and Ayvaz[92] have performed SA using Twitter data for Turkish and English language. RSentiment tool was used to classify English tweets while Turkish tweets were classified using dictionary based approach with manually created lexicon. It was observed that Turkish tweets carry more positive sentiments about Syrians and refugees compared to English tweets. Sentiment of Turkish tweets were evenly distributed across positive, negative and neutral classes.

While English tweets largely bear neutral and negative sentiments. Garg et al.[33] study the temporal distribution of sentiment using ensemble of Naive-Bayes and SVM algorithms on Twitter data in English language. This study however, have not discussed the dataset creation approach or details of the dataset used for the study.

2.4 Summary

The sentiment analysis studies have been carried out in three directions, i.e., sentiment feature extraction, sentiment classification, and application of sentiment analysis. Most of the recent techniques of sentiment analysis are based on deep learning methods. However, performing sentiment analysis using deep learning methods requires a large collection of resources. Therefore, most of the existing studies of sentiment analysis are inclined to the customer reviews domain. The application of sentiment analysis on the societal domain uses one of the available off-the-shelf tools to extract sentiment information for quantification and summarizing the user views on various societal issues. However, the effectiveness of using the off-the-shelf sentiment analysis tools is subject to investigation. In the following chapters (Chapters 3 and 4), the characteristics of opinions on the societal and non-societal domains are investigated and performed an empirical study to assess the efficacy of the sentiment analysis tools and techniques over the societal domain.

There are things known and there are things unknown, and in between are the doors of perception.

Aldous Huxley, English writer

3

Characteristics of opinions on societal and non-societal datasets

This study investigates the characteristics of societal and non-societal datasets through statistical analysis of text and network representations. The text-based analysis investigates the corpus and word usage characteristics across a wide range of domain and topics. On the other hand, the graph-based analysis aims to uncover the global properties of the words in the domain dataset. In this study, so-

cietal topics are defined as topics or events related to social unrest, terrorist acts, or government policies. General opinions, product reviews, movie reviews, and restaurant reviews, on the other hand, are considered non-societal topics. This research utilized an in-house curated societal dataset and online available customer review datasets such as product reviews and movie reviews as non-societal datasets to perform statistical analysis. From various experimental investigations, it is observed that the vocabulary used in the product reviews and movie reviews datasets have similar sentiment word associations. In comparison, the vocabulary in the societal and consumer review datasets did not share any sentiment word associations. It is also observed that opinions on Twitter adhere to scale-free network properties, increasing the possibility of using social network analysis techniques to investigate sentiment analysis studies from a network perspective.

3.1 Introduction

Sentiment analysis research has gained much importance as user-generated content and social media platforms have grown rapidly since early 2000[95,123]. In the last two decades, a number of researchers have made significant contributions to the field of sentiment analysis. Majority of the sentiment analysis studies focus on customer review domains like product reviews and movie reviews compared to the societal domain. Primarily, this is due to the wide range of societal topics and geographic differences in the types of social issues. However, with the increasing active user participation in expressing their views on various societal issues on social media platforms, sentiment analysis of public opinions has been increasingly important for various agencies such as data analysts, social scientists, corporate,

government departments, etc.

The primary objective of sentiment analysis is to identify the sentiment expressed in a given piece of opinionated text. The sentiment of an opinionated text can be classified using machine learning models or rule-based heuristic algorithms. However, developing a sentiment analysis classifier requires a significant amount of annotated datasets and domain expertise (in the case of a rule-based classifier). Although there is a significant amount of annotated corpus for customer review domains such as product reviews, movie reviews, and hotel reviews, no gold standard dataset exists for the societal domain. Moreover, the sentiment analysis task is highly domain-dependent, i.e., a sentimental word in one domain can be the opposite of sentiment in another domain[94,46,62,63,35,104]. Therefore, to benefit from the rich resources of the non-societal domain over the low-resource societal domain, it is crucial to analyze the characteristics of opinions and the association of vocabularies between the societal and non-societal domains.

The characteristics of opinions on societal and non-societal domains are investigated in this study via statistical analysis of text and network views using various datasets covering societal and non-societal domains. This study first investigates the word distribution across the corpus via text-based analysis to evaluate if it adheres to the Principle of Least Effort* using Zipf's and Heap's law. The semantic association of words across various domains is investigated via the Pointwise Mutual Information (PMI) method[24] to understand the similarity pattern of the vocabularies association across different corpus or topics. Finally, the similarity of a corpus over another corpus of different topics and homogeneity within the corpus are analyzed by measuring the perplexity of the opinions based on language

*https://en.wikipedia.org/wiki/Principle_of_least_effort

models constructed from various corpora.

This study further investigates the characteristics of word relations in a domain using graph-based analytic methods by transforming the corpus into a word co-occurrence graph. The strength of the word relations is computed over the transformed co-occurrence graph using the clustering coefficient method to determine if the graph comprises weak or strong ties. Furthermore, the word associations in the graph are evaluated by measuring whether the words are clustered together or appear discretely by finding the connected components of the graph. Finally, the co-occurrence graph is analyzed to see if it follows a scale-free network characteristic by calculating the exponent of the power-law distribution over the node degree distribution. To study the characteristic difference between societal and non-societal datasets, an in-house curated `Societal` dataset is considered while the online available customer review datasets namely product reviews posted in Amazon*, Twitter†, and movie reviews[95] posted in IMDb‡ are considered as non-societal datasets. Table 3.1 shows the characteristics of the datasets considered in this study.

3.2 Experimental Setup

3.2.1 Datasets

Dataset preparation - Societal dataset

This section discusses the curation process of the in-house dataset named `Societal`. We manually identified popularly used event-specific hashtags in order to collect

*www.amazon.com
†http://help.sentiment140.com/for-students/
‡https://www.imdb.com/

Table 3.1: Characteristics of the Experimental Datasets

Dataset	Pos	Neg	Neu	Total	Topics	Domain
Soceital	17,304	19,869	9705	46,878	Kashmir Unrest, Pathankot Attack, Surgical Strike, GSTN*, Demonetization, Uri Attack, Paris Agreement, Syria Crisis	Social Issue
– Kashmir Unrest	1363	3638	947	5948	–	Social Issue
– Pathankot	1044	3722	1039	5805	–	Social Issue
– Surgical Strike	2116	3278	2191	7585	–	Social Issue
– GSTN	11852	6409	4823	23084	–	Social Issue
– Demonetization	653	1540	126	2319	–	Social Issue
– Uri Attack	126	416	205	747	–	Social Issue
– Paris Agreement	83	149	147	379	–	Social Issue
– Syria Crisis	67	717	227	1011	–	Social Issue
SemEval-2016	1296	2491	276	4063	Atheism, Climate Change, Feminist Movement, Hillary Clinton, Legalization of Abortion	Social Issue
Sentiment-140[$]	799978	800024	–	1600002	Consumer reviews discussion	Product Review
Amazon[!]	2000000	2000000	–	4000000	Consumer reviews discussion	Product Review
Movie Review[+]	1000	1000	–	2000	Movie reviews discussion	Movie Review

[$] Dataset downloaded from http://help.sentiment140.com/for-students/
[!] Dataset downloaded from https://www.kaggle.com/bittlingmayer/amazonreviews
[+] Dataset downloaded from https://www.cs.cornell.edu/people/pabo/movie-review-data/

tweets[†] of the events from Twitter. Using the Twitter Streaming API[‡], we were able to crawl 50,300 tweets. Two annotators have been assigned to these tweets to annotate the sentiment (i.e., positive, negative, or neutral). The languages of interest for annotating tweets are English and code-mixed Hindi and English. Both the annotators are fluent in both English and Hindi. As a guideline for annotation, the annotators are briefed to annotate the tweets based on textual content, without considering event context such as entities engaged, tweet author information, and so on. For example, people who support the event Surgical strike may express positive sentiment tweets. However, those who opposed the event can also express negative sentiment tweets. Since the event is about attacking people, tweets with such characteristics are annotated as negative sentiment. The annotators agree on the exact sentiment of 46,878 out of 54,550 tweets, with an 82.35 Kappa coefficient. According to the annotator's judgment, majority of the tweets on societal topics have sentiment polarity while only a few tweets are objective,

[†] Opinionated text in Twitter
[‡] http://docs.tweepy.org

i.e., few tweets with neutral sentiment. The majority of tweets with disagreement are a consequence of the annotators' judgment of neutral sentiment. The same characteristics have also been reported in the study of Maynard et al.[73]

SemEval-2016

This dataset was created as the challenge dataset for the SemEval-2016 Stance detection task by Saif et al.[81]. The authors performed sentiment analysis on this dataset and achieved the best performance up to 76.4 F-macro scores by leveraging an inhouse curated sentiment lexicon[82] as features. This thesis work considers using this lexicon for word correlation and association analysis.

Amazon product reviews

McAuley et al.[74] curated this dataset for product recommendation tasks based on product reviews and ratings. The product reviews are based on laptop, movies, and books available on the Amazon website*. This dataset has been used for various text-classification[146] and sentiment classification[51,147] tasks.

Sentiment-140

Go et al.[36] curated this dataset for distant supervision sentiment analysis of tweets using emoticons. The dataset was filtered using phrases based on product or movie names such as Visa, Star Trek, Nike, etc.

*https://www.amazon.com/

Movie reviews

This dataset was curated from the Internet Movie Database (IMDb)* by Pang et al.[95] for sentiment analysis. This dataset was also used in Maas et al.[72] study for word representation learning on the sentiment analysis task.

3.2.2 Text analysis methods

The objective of the text-based analysis study is to understand the characteristics of word usage and corpus similarity across societal and non-societal domains.

Word distribution analysis

According to the Principle of Least Effort, human nature desires the maximum benefit for the least effort (word usages). The statistical characteristic of word distribution across datasets is investigated using Zipf's and Heap's laws to determine if the considered corpora follow natural phenomena or the vocabularies of the corpus keep evolving due to numerous user associations.

Zipf's Law states that the rank r of a word with frequency f in the corpus approximately follows the equation:

$$f(r) \propto cr^{z} \tag{3.1}$$

where c is a constant number and r is the rank based on the frequency, denoted as $f(r)$ and z is approximately equal to 1. That is, the second rank word has half the occurrences of the first rank word, the third rank term has one-third of the first, and so on. A log-log graph plot of a term's frequency as a function of its rank is

*https://www.imdb.com/

identically a line with slope $z = -1$, as provided by the power-law equation:

$$log(f(r)) = log(c) + zlog(r). \tag{3.2}$$

Heap's Law represents vocabulary size M as a function of collection size:

$$M = kT^b \tag{3.3}$$

where T is the total number of words occurrences in the collection, k and b are parameters. According to Heaps' law, as more text instances are accumulated, the possibilities of uncovering a widespread vocabulary from which the individual tokens are derived decreases. The motivation for Heap's law is that the simplest possible relationship between collection size and vocabulary size is linear in log-log space, as in Zipf's Law. The heaps law for corpus `Reuters-RCV1` gives a slope of 0.49 and intercept = 1.64*.

Association of words across domains analysis

Pointwise Mutual Information (PMI)[24] is used to analyze the semantic associations of words across various corpora[123,124]. PMI is a quantitative measure of the co-occurrence of an event (presence or absence), such as the presence of a word in a corpus or the co-occurrence of tokens in a corpus. Mutual Information (MI) may also be used to assess how much information the presence and absence of a term contributes to the corpus under consideration. MI is the expected value or average of the PMI scores for the presence or absence of a word in the corpus. This study

*http://nlp.stanford.edu/IR-book/html/htmledition/heaps-law-estimating-the-number-of-terms-1.html

considers analyzing the semantic associations of the words over the considered corpora using the PMI method. Equation 3.4 defines the mathematical formula for finding PMI of a term t appearing in a corpus c.

$$PMI(t;c) = \log \frac{P(t/c)}{P(t)} \tag{3.4}$$

where $P(t/c)$ is the conditional probability of token t appearing in corpus c. $P(t)$ is the probability of token t in the considered corpora. PMI can also be used to find the semantic orientation of two tokens in a corpus. Equation 3.5 defines the mathematical formula for finding PMI of a term t_1 co-occuring with term t_2 in a corpus.

$$PMI(t_1,t_2) = \log \frac{P(t_1,t_2)}{P(t_1)P(t_2)} \tag{3.5}$$

where $P(t_1,t_2)$ defines the probability of tokens t_1 and t_2 co-occur, $P(t_1)$ and $P(t_2)$ is the probabilities of individual tokens in a corpus. The ratio of the PMI score defines the statistical dependency of the two tokens in a corpus.

The strength of word association with sentiment lexicon can be analyzed using the PMI score of words co-occurring with sentiment polarized words in a corpus[124]. The strength of word association with sentiment lexicon is calculated as follows:

$$SOA(w_i) = \sum_{\forall w_p \in Positive\ \ set} PMI(w_i,w_p) - \sum_{\forall w_n \in Negative\ \ set} PMI(w_i,w_n) \tag{3.6}$$

Here the *Positive* and *Negative* sets are the group of words from a publicly available sentiment lexicon of the respective sentiments. Word w_i is said to have positive semantic orientation when the score of $SOA(w_i)$ is positive otherwise it is said to have negative semantic orientation.

Homogeneity and similarity of corpus analysis

A corpus is similar to itself (homogeneous) if the language in the corpus does not vary. Likewise, a corpus is comparable to another corpus if the language constructs are similar[47]. A language model can be used to estimate the likelihood of language constructs within a corpus or between corpora. The language model is a statistical model that assigns probabilities to words and sentences using probability distributions learnt from training corpora. Sentences that are real and syntactically align to the training corpus of the language model will have a high probability score. Using a statistical n-gram based language model (n = 3 in this study), the probability of a sequence of words ($\mathbf{W} = (w_1, w_2, ..., w_N)$) can be defined as:

$$P(STRT, STRT, w_1, w_2, \cdots, w_N, END) = \prod_{k=1}^{N+1} P(w_k | w_{k-1}, \cdots, w_{k-n-1}) \tag{3.7}$$

where (w_{-1}, w_0) and w_{N+1} are the $STRT$* and END tags added to every sentence while training the language model.

Various studies have considered perplexity as an intrinsic evaluation metric for assessing language model[47,108]. A language model (LM) with a lower perplexity score determine a better language model. Perplexity of a language model can be define as:

$$PP(W) = 2^{-\frac{1}{N} log_2 P(W)} \tag{3.8}$$

By measuring the perplexity of the language models while keeping the language model constant, we can assess the homogeneity and similarity of corpora.

*n-1 number of $STRT$ tags are added at the beginning of the sentence.

The *homogeneity of a corpus* can be determined by training a language model over the corpus and evaluate the language model perplexity over the same corpus's testing set. A corpus is not homogeneous if the perplexity score is high, indicating that the language used in the corpus varies significantly. On the other hand, the *similarity of corpora* can be estimated by training a language model on one corpus and evaluating the perplexity on the testing set of another corpus. A corpus is not similar if the average perplexity score is high, indicating that the language used in one corpus differs from the language used in another.

3.2.3 Graph analysis methods

The characteristics of the datasets are analyzed from a network analysis perspective by representing each dataset in a graph structure. This analysis aims to understand the word relations regardless of the language construct used in the corpora. If the words are strongly clustered, it indicates that their relationship follows a regular syntactic convention. If the relations are disjoined or weakly clustered, it indicates that word relations are not uniform and possibly from various languages or topics.

Representing corpus in a graph structure

The language we use to express ourselves may be represented as a network of words connected through grammatical relationships. In recent times, while expressing an opinion on social media platforms such as Twitter, users often use hashtags and mentions to reflect meta-information such as sentiment, emotion, topic, or entity and draw the attention of the mentioned users to the user's opinions. To accommodate the relations of keywords (K), hashtags (H), and mentions (M),

this study consider a multi-layer network $\mathbf{G} = (\mathbf{V}, \mathbf{E}, \mathcal{L})$ with $\mathcal{L} = 3$ layers. The network consists of both directed and undirected edges to capture both the co-occurrence and sequential characteristics of K, H, and M in a tweet. An edge $e_{x,y} \in \mathbf{E}$ is directed if x and y occur sequentially next to other in a tweet where, i) $x, y \in K$ or ii) $x \in K$ and $y \in \{H \cup M\}$ or iii) $x \in \{H \cup M\}$ and $y \in K$. Whereas, an edge $e_{x,y} \in \mathbf{E}$ is undirected if $x, y \in \{H \cup M\}$ co-occur in a tweet. An example of the multi layer network for the tweet "*Historic day for the Nation, #GST bill passed in Lok Sabha. #Congratulations to the nation,salute 2the vision of #PM @narendramodi ji*" is shown in Figure 3.1. This multi-layer network have three types of intra-layer associations $\mathbf{A} = \{\mathbf{A}^K, \mathbf{A}^H, \mathbf{A}^M\}$ and five types of bipartite associations $\mathbf{B} = \{\mathbf{B}^{HM}, \mathbf{B}^{MK}, \mathbf{B}^{HK}, \mathbf{B}^{KM}, \mathbf{B}^{KH}\}$ where $\mathbf{A}^i \in \mathcal{R}^{N^i \times N^i}$ is the adjacency matrix in layer $i \in \{K, H, M\}$, $\mathbf{B}^{i,j} \in \mathcal{R}^{N^i \times N^j}$ is the inter-layer relation between layer i and layer j, and N^i is the number of nodes in the respective layers. This network can also be viewed as one flattened representation in form of supra-adjacency matrix S, with total nodes $N = |\mathbf{V}^H| + |\mathbf{V}^M| + |\mathbf{V}^K|$,

$$\mathbf{S}_{N \times N} = \begin{bmatrix} \mathbf{A}^H & \mathbf{B}^{HM} & \mathbf{B}^{HK} \\ \mathbf{B}^{MH} & \mathbf{A}^M & \mathbf{B}^{MK} \\ \mathbf{B}^{KH} & \mathbf{B}^{KM} & \mathbf{A}^K \end{bmatrix} \tag{3.9}$$

The intra-layer associations $\mathbf{A}$s are on the main-diagonal, and the cross-layer connections $\mathbf{B}$ are on the off-diagonal elements of $\mathbf{S}$. Further, $\mathbf{A}^K, \mathbf{B}^{HK}, \mathbf{B}^{KH}, \mathbf{B}^{MK}, \mathbf{B}^{KM}$ are asymmetric matrices and other matrices of $\mathbf{S}$ are symmetric. In similar fashion a tweet or a collection of tweets can be represented as a multi-layer network.

Tweet:*Historic day for the Nation, #GST bill passed in Lok Sabha. #Congratulations to the nation,salute 2the vision of #PM @narendramodi ji*

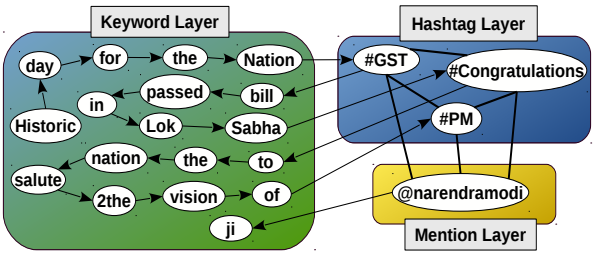


Figure 3.1: An example of representing a tweet to a heterogeneous multi-layer network structure.

Clustering Coefficient

Clustering Coefficient (CC) is a measure of how strongly nodes in a network are clustered. It assesses the ego network* property to estimate the likelihood of a node being associated with another. The CC is computed by measuring the density of the subgraphs that remain connected after eliminating ego and the edges that are incident on ego. The CC can be categorized into two versions, namely global and local. The global version depicts the network's overall clustering, whereas the local version depicts the cohesiveness of individual nodes. This study aims to evaluate if the word associations in the graph are of weak or strong ties using the average estimates of local clustering coefficients for selected sentiment-oriented seed nodes in the graph. Given a graph $G = (V, E)$ with V nodes and E edges, the local clustering coefficient of a node (C_i) can be define as:

$$C_i = \frac{|\{e_{jk} : v_j, v_k \in N_i, e_{jk} \in E\}|}{k_i(k_i - 1)} \tag{3.10}$$

*A subgraph based on the connection of one central node known as the ego in a graph.

where Ni and ki denote the set of neighboring nodes and the number of neighboring nodes of ego i, respectively. The average clustering coefficient is the average of the local clustering coefficient scores of the sentiment seed nodes in the graph G.

Connected components

A connected component (or simply component) is a network subgraph that is disconnected from other components. In a network, there can exist multiple components. Among the components, there exists a giant component where a significant amount of the nodes in the network are connected. The purpose of this study is to investigate if word associations in vocabularies are isolated or clustered, regardless of whether the associations are weak or strong. If the network has many components, it implies that the word associations in the individual components are related to a comparable syntactic word convention.

Scale free network analysis

A scale-free network is defined as one that asymptotically follows a power-law degree distribution. Any real-world network can be interpreted as power-law degree distributions, such as follower-followee networks in social networks like Twitter and Instagram, airway and railway routes, and so on. Since the language we use to express ourselves is a network of words linked together through syntactic relationships, in this study, we would like to investigate if the opinions follow a scale-free network property. The degree distribution of a network having k nodes can be defined as follows:

$$P_{deg}(k) = k^{-\gamma} \tag{3.11}$$

Table 3.2: Slopes and intercepts of Zipf and Heap plots.

Dataset	Zipf		Heap	
	Slope	Intercept	Slope	Intercept
Societal	−0.651	−5.591	0.646	2.312
SemEval 2016	−0.351	−2.495	0.787	1.087
Sentiment140	−0.478	−5.322	0.714	1.812
Amazon	−0.777	−9.167	0.691	3.150
Movie	−0.966	−7.911	0.513	3.684

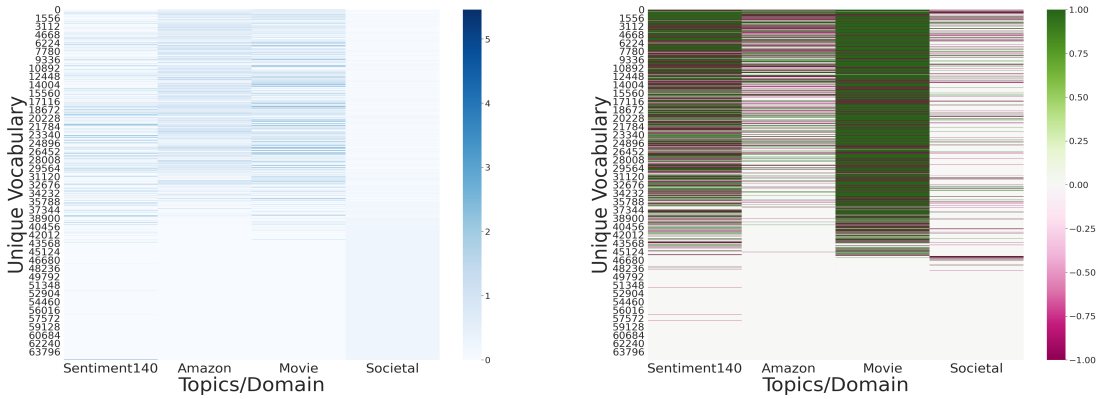
where γ is a parameter typically in the range $2 < \gamma < 3$ for a scale-free network. The function $P_{deg}(k)$ decays slowly as the degree k increases.

3.3 Observations

3.3.1 Text analysis

This study first analyses the word distribution across the corpus using text-based analysis and Zipf's and Heap's plots. Table 3.2 summarises the slopes and intercepts of Zipf and Heap log-log plots for the societal and non-societal datasets (i.e., `Societal`, Sentiment140, Amazon, and Movie reviews)*. It is evident from the table that the slope of Zipf's plot for the Movie review dataset is closer to −1, indicating that it firmly follows the Principle of Least Effort. Furthermore, the slope of Heap's plot is nearly 0.5, indicating that the movie review dataset has almost completely covered the corpus's word distribution. In contrast, the slope of Zipf's plots for Amazon review dataset and Twitter datasets such as `Societal`, Sentiment140 are far from −1, indicating that the opinions are noisy due to misspelling, creative writing, usage of slang, and so on, and the Principle of Least Effort is followed minimally. Furthermore, the slopes of Heap's plots across these corpora are higher than 0.5, indicating that the corpora have not yet covered the

*SemEval-2016 dataset is excluded from this study because of the small corpus size.



(a) Pointwise mutual information

(b) Strength of sentiment word association

Figure 3.2: Heatmap plot of word vocabularies information in societal and non-societal datasets.

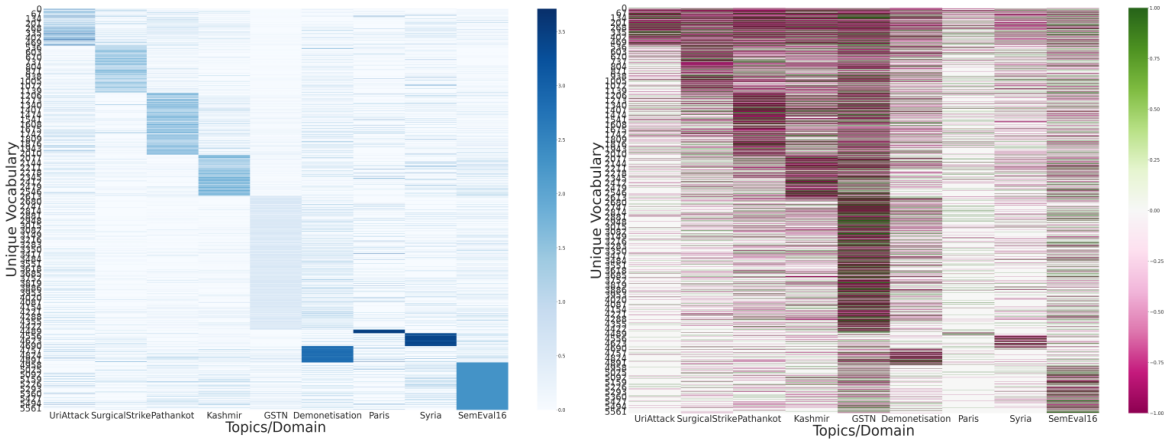
corpus's vocabulary entirely. This study shows that the opinions in the Movie reviews dataset are written in a more structured manner than the opinions in the `Societal`, Sentiment140, and Amazon corpora.

The following study investigates the word association concerning the topics and its sentiment word association using PMI and the SOA. The heatmap plots of PMI and SOA scores for the top occurring tokens across societal and non-societal datasets is shown in Figure 3.2. Figure 3.2 (a) shows the distribution of the tokens with high information content for the societal and non-societal datasets. The figure indicates negligible overlapping of informative tokens between the societal and non-societal datasets. It indicates that the informative tokens in social and non-societal domains have different meanings and the informative tokens associated with non-societal datasets have similar informative contents. Further, Figure 3.2 (b) shows the strength of association between the above informative tokens and a seed sentiment lexicon. It is evident from the figure that informative tokens of non-societal datasets have a higher strength of association with the seed words

in the sentiment lexicon than that of the societal dataset. For instance, the tokens like *#ModiPunishesPak*, *#IndiaStrikesBack*, *#UriAttack*, *#DeMonetisation*, and *#KashmirUnrest* (less sentiment expressive tokens) have higher information content in the societal dataset, whereas the token like *beautiful*, *hate*, *best*, and *soulful* (higher sentiment expressive tokens) have high information content in the non-societal datasets.

Since the `Societal` dataset includes various topics such as *Uri attack*, *Pathankot attack*, *Surgical strike*, etc., this study further investigates the word similarities associated with these topics. Figure 3.3 shows a heatmap visualization of PMI and SOA scores for the most commonly occurring tokens in the `Societal` dataset across the same wide range of topics. Figure 3.3(a) indicates how each topic has different word associations that may better represent the topics based on the PMI distribution. It is also worth noting that the topics of similar themes, such as *Uri attack*, *Pathankot attack*, *Surgical strike*, and *Kashmir unrest*, have similar word associations. Figure 3.3(b) indicates that majority of the tokens are significantly linked with negative emotion. The vocabulary used in topics related to the Indian context has a semantic orientation similar to sentiment tokens. It is evident from this study that the vocabulary used in the `Societal` dataset has a weak semantic orientation to sentiment tokens in contrast to consumer review datasets. Furthermore, it is observed from Figure 3.3 (b) that topics of the related themes have shared similar vocabulary with the same semantic orientation of sentiment tokens among the `Societal` topics.

Finally, the homogeneity and similarity of corpora are evaluated via an intrinsic evaluation of a language model (LM) based on the perplexity score using a 10-fold cross-validation approach. The homogeneity of each corpus is evalu-



(a) Pointwise mutual information

(b) Strength of sentiment word association

Figure 3.3: Heatmap plot of word vocabularies information of societal topics.

Table 3.3: Corpus homogeneity and similarity of corpora using perplexity score.

	Dataset	Societal	Sentiment140	Amazon	Movie
LM	Societal	16.32 (±2.07)	20.09 (±3.90)	17.33 (±0.91)	17.38 (±0.40)
	Sentiment140	20.21 (±2.24)	17.38 (±3.74)	16.25 (±1.19)	16.98 (±0.55)
	Amazon	20.26 (±2.20)	16.30 (±4.19)	15.37 (±1.11)	16.33 (±0.53)
	Movie	20.30 (±2.18)	16.38 (±4.15)	16.52 (±0.95)	15.50 (±0.52)

* LM: Language model

ated using the average perplexity score of its 10 LMs. Since the LMs are trained using a 10-fold cross-validation approach, the similarity of the two corpora is computed by averaging the perplexity scores of the ten trained LMs over the ten testing sets of another corpus. Table 3.3 shows the average perplexity score of the language models for each corpus across the entire corpora testing set. As shown in the diagonal components of the table, the average perplexity score of the Amazon product (15.37) and Movie (15.50) reviews datasets are lower than the `Societal` (16.32) and Sentiment140 (17.38) datasets. It implies that the Amazon and Movie reviews datasets are more homogeneous than the `Societal` and Sentiment140 datasets. On comparing the corpus similarity of the `Societal` dataset to the rest, it is observed that the LMs' average perplexity scores across

these datasets, i.e., Sentiment140 (20.09), Amazon (17.33), and Movie reviews (17.38), are higher than their own (16.32). It implies that the `Societal` dataset is different from these non-societal datasets, with Sentiment140 corpus being the most different. Similarly, using LMs trained on the Sentiment140 dataset, it is observed that the perplexity score of the LMs over Sentiment140 dataset (i.e., 17.38) is higher than the Amazon (i.e., 16.25) and Movie (i.e., 16.98) datasets. It indicates that the Sentiment140 dataset is similar to the Amazon and Movie reviews datasets. However, the `Societal` dataset has a higher perplexity score than the Sentiment140 dataset, revealing that the language constructs used in the `Societal` dataset are not similar to those used in the Sentiment140 dataset. Similarly, it is also observed that using the LMs trained with the Amazon (15.37) and Movie (15.50) reviews datasets, the perplexity score over the `Societal` dataset is more than 20, and the Sentiment140 dataset has a perplexity score of roughly 16.30. This study clearly shows that the language construct used in the `Societal` dataset differs from that of the non-societal datasets.

3.3.2 Graph-based analysis

In this section, the properties of the considered corpora are investigated from a network analysis perspective by representing each corpus in a graph structure (discussed in Section 3.2.3). One advantage of transforming a tweet to a graph structure is that it circumvents the need for language-specific analysis. Table 3.4 summarizes different network characteristics such as node statistics, number of connected components, and number of nodes belonging to giant connected components for all corpora considered in this study. The statistics show that opinions posted on Amazon and IMDb (movie reviews) use fewer hashtags and mentions

Table 3.4: Characteristics of the type of network representation of societal and non-societal datasets

	Societal	SemEval-2016	Sentiment-140	Amazon	Movie
Unique Vocabulary	50,184	11,468	605,284	2,669,763	39,969
Hashtags	10.55%	22.13%	1.44%	0.35%	0.05%
Mentions	11.05%	9.97%	51.16%	0.15%	0.03%
Keywords	78.40%	67.90%	47.39%	99.50%	99.93%
Edges	238,818	56,049	2,825,303	40,008,960	470,718
Degree$_{max}$	15,259	11,062	66,739	2,115,792	12,486
Degree$_{mean}$	15.753	23.267	282.284	1670.221	28.465
Degree$_{min}$	1	2	1	1	1
CC	100	10	11	13	1
GC	99.45%	99.67%	11.03%	79.25%	100.00%
Power_law$_{exponent}$	1.790	1.755	1.292	1.245	1.320

* **CC**: Connected Component, **GC**: Percentage of nodes belonging to Giant CC

than those posted on Twitter (Societal, SemEval-2016, and Sentiment140). It could be due to the fact that hashtags and mentions are less popular on these platforms while curating these datasets. Further, the Twitter datasets have a large number of connected components, with Societal having the highest connected components. Except for product review datasets (Sentiment-140 and Amazon reviews), almost all the nodes of the considered datasets belong to giant connected components, which is a desirable property for real-world social and information networks analysis. Furthermore, the Power_law$_{exponent}$ score for Societal, SemEval-2013, and SemEval-2016 is closer to 2, indicating that these networks adhere to scale-free network features*. It highlights how a small number of tokens (or nodes) are predominantly utilized (or connected) with the remaining nodes, which is intuitive in most real-world social and information networks. This analysis paves the way for numerous social network analysis studies that can be performed on this tweet graph.

This study further investigate the node properties through local clustering

*https://en.wikipedia.org/wiki/Scale-free_network

Table 3.5: Average clustering coefficient of sentiment tokens in the word graph

Datasets	Positive	Negative
Societal	0.140 (±0.23)	0.141 (±0.22)
SemEval-2016	0.302 (±0.35)	0.312 (±0.35)
Sentiment140	0.290 (±0.23)	0.302 (±0.22)
Amazon	0.462 (±0.20)	0.472 (±0.19)
Movie	0.439 (±0.30)	0.473 (±0.31)

coefficient measures to understand if the considered sentiment lexicon have strong association in the tweet graph. Table 3.5 shows the average clustering coefficient scores of the sentiment words over the considered datasets. It is observed that the Amazon and Movie review datasets have better average clustering coefficient of above 0.4 than the rest of the datasets. This indicates that the sentiment words are better utilized in such platforms than in Twitter. Among the Twitter datasets, it is observed that the `Societal` dataset has the lowest average clustering coefficient (0.14). This implies that the language used in `Societal` dataset is different from the language of the sentiment lexicon.

3.4 Summary

This article uses text and graph-based analysis to examine the features of opinions on societal and non-societal datasets. The Zipf and Heap plots of the text-based statistical analysis show that the Twitter datasets do not follow the Principle of Least Effort. Further, the PMI analysis indicates that the customer review datasets shared most of the tokens associated with other customer reviews datasets considered in this study. In contrast, it is observed that the `Societal` dataset has little or no word association with the customer review datasets. Among the various topics in the societal domain, similar topics share a strong association of

tokens, and each topic has its own set of distinctive characteristics. The lexical distribution of the datasets reveals that hashtags are used less often in the customer review domains. In comparison, the `Societal` and SemEval datasets contain more than 10% coverage of hashtags over the entire vocabulary of the dataset. It shows that hashtags are used commonly while expressing opinions on Twitter. Furthermore, network analysis unveils that the network representations of the `Societal` and SemEval datasets adhere to scale-free network features, i.e., the word graphs adhere to real-world network structure. It shows that sentiment analysis of tweets may be investigated from the perspective of network representation in addition to text-based techniques.

No man can hope to find out the truth without investigation.

George F. Richards

4

Empirical Study of Sentiment Analysis Tools and Techniques on Societal Topics

In recent times, a surge in public opinion mining against various societal topics using publicly available off-the-shelf sentiment analysis tools is evident. Since sentiment analysis is a domain-dependent problem, the sentiment analysis tools available online are mainly for customer reviews. Therefore, the suitability of using such existing off-the-shelf tools for a societal topic is subject to investigation.

There exist no such studies that have thoroughly investigated the effectiveness of using off-the-shelf tools on societal issues. This study systematically evaluates the performance of 10 popularly used off-the-shelf tools and 17 state-of-the-art machine learning techniques and investigates their strengths and weaknesses using various societal and non-societal topics.

4.1 Introduction

With the increase in availability of public opinions on various social media platforms such as Twitter, Facebook, LinkedIn, Google Plus, YouTube, etc., a surge in attention of data scientists/agencies in understanding public opinions on various social issues such as social inequality[16,116,59], public health[46,73,32], social unrest[92], election[55,107,83,122], disaster events[87,22], terror attack[15], etc. is evident. Understanding public opinion on various social issues is vital for various communities like business associates, policymakers, law enforcement agencies, etc. One of the parameters often considered in such studies is public sentiment toward target policies or issues. As building a sentiment analysis (SA) tool is an expensive task that potentially needs a large volume of annotated dataset and domain expertise, most of the studies that analyze public opinion use publicly available off-the-shelf tools. However, it is observed from various studies[104,73,35,112,149] that the task of SA is highly domain-dependent. A SA tool built for product reviews may not be suitable for finding sentiment of public opinions in the societal domain and vice versa. Therefore, the effectiveness of using off-the-shelf tools for SA on public opinion over various societal topics needs systematic investigation. Motivated by the above observations, this study systematically evaluates the performances of

publicly available SA tools over various datasets collected from Twitter in the domain of social issues, product reviews, movie reviews, and restaurant reviews.

Researchers have evaluated responses of off-the-shelf publicly available sentiment analysis tools like SentiStrength*, Sentiment-140†, RSentiment‡, EmoLex§, Vader¶, etc. in the past [104,73,1,38]. However, these evaluations mostly consider datasets from domains like products review, movies review, etc. For instance, authors in [104,1] have evaluated a broad set of publicly available sentiment analysis tools over the customer reviews/comments (products, movies, news articles, Youtube videos). It is reported in these studies that most of the publicly available tools respond differently to datasets of different domains. Recently, many of the data scientists use such off-the-shelf publicly available tools to analyze public sentiment over various societal topics without justifying the underlying tools' effectiveness. For example, studies in [22,55,87,59,73,107] have used SentiStrength to study public sentiment against topics like political analysis, natural disaster, climate change, multilingual polarity, etc. Singh et al. [116] have used MeaningCloud‖ for analyzing public opinion related to government policies. Studies in [92,90] have used RSentiment to analyze public sentiment on social unrest issues and childhood vaccination. Considering the volume of such studies, a systematic evaluation of these tools in the domain of societal topics is warranted. Except in [107,73,38], none of the existing studies have considered societal issues to the best of our knowledge to evaluate the off-the-shelf sentiment analysis tools. Though authors in [107,73,38]

*http://sentistrength.wlv.ac.uk
†http://www.sentiment140.com/
‡https://cran.r-project.org/web/packages/RSentiment/index.html
§http://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm
¶https://www.nltk.org/_modules/nltk/sentiment/vader.html
‖https://www.meaningcloud.com

have considered few selected societal topics, they have not cross-evaluated responses of the tools across opinions on societal issues and product/movie reviews. Motivated by the above research gaps, this study revisits the evaluation task, and systematically evaluates various popularly used off-the-shelf publicly available sentiment analysis tools to study the suitability of using them while analyzing public opinion on societal topics.

4.1.1 Research goal and contributions

The primary goal of this study is to investigate the suitability of using publicly available off-the-shelf sentiment analysis tools for determining public sentiment over societal issues. In particular, it attempts to answer the following research questions:

1. Are the off-the-shelf sentiment analysis tools suitable for finding public sentiment over societal topics?
2. Are the sentiment classifiers built over societal and non-societal topics compatible?
3. Can we generalize a sentiment classifier built over societal topics across different geographical locations?

To answer the above questions, we evaluate ten popularly used publicly available sentiment analysis tools over various datasets created from societal and non-societal domains. To understand the cross-domain and generalization characteristics of a sentiment classifier under societal topics, we further locally build 17 different sentiment classifiers using state-of-the-art machine/deep learning methods

over different datasets of societal and non-societal domains. For ease of reference, we refer to the locally built classifiers as `Techniques` and the off-the-shelf tools as `Tools` in the rest of this chapter. This study covers a total of twenty-seven (27) classifiers (17 `Techniques` and 10 `Tools`) and eight different datasets of different natures/domains, and make the following observational contributions:

- This study systematically investigates the suitability of using publicly available off-the-shelf sentiment analysis tools for analyzing public sentiments over societal topics.
- It is observed from various experiments that majority of the off-the-shelf tools are biased towards customer reviews, and not suitable for the societal domain.
- Sentiment classifiers are not cross-domain compatible between societal topics and customer reviews.
- Public opinions on general societal topics have generalization characteristics through shared sentiment bearing words across different countries.
- Neural network-based classifiers can capture better textual characteristics than feature-based classifiers.
- Due to unavailability of suitable off-the-shelf sentiment classifier for societal topics, locally built classifiers dominate most of the publicly available sentiment analysis tools.
- Strengths and weaknesses of different Tools and Techniques are briefly analysed over different sentiment analysis sub-tasks such as code-mixed text,

sarcastic comments, text with aspect and stance etc.

Sections of the chapter are organized as follows. Section 4.2 presents a brief review of the literatures relevant to this chapter. In Section 4.3, the experimental setup for this study is presented. Experimental observation and error analysis are reported in Section 4.4. Finally, in Section 4.5, we summarized the study of this chapter.

4.2 Related studies

This section briefly discusses studies related to (i) application of SA `Tools` on various public opinion analysis, (ii) evaluation of the publicly available SA `Tools`, and (iii) evaluation of the state-of-the-art SA `Techniques`.

4.2.1 Usage of Publicly Available Sentiment Analysis Tools in Social Media Data Analysis

Most of the recent studies on social media data analysis use off-the-shelf publicly available sentiment analysis tools for analyzing public opinions. To name a few, authors in [22,16,87,59] have studied the spatio-temporal sentiment pattern in various regions of the United States of America (USA). Chen et al.[22] have used SentiStrength tool to quantify the sentiment distribution of the affected and unaffected regions in Texas* during Hurricane Harvey†. Cao et al.[16] have used IBM Watson Alchemy API to quantify the sentiment of people in the region of the study. They observe the distinctive characteristics of people's sentiment across different land use and time. Neppalli et al.[87] have studied sentiment distribution

*One of the state of USA

†https://en.wikipedia.org/wiki/Hurricane_Harvey

of public opinions on the disastrous event Hurricane Sandy*. They use two binary classifiers for classifying neutral, positive, and negative sentiments. For classifying neutral or subjectivity, they use SentiStrength, and for polarity classification, they have used the SVM classifier. Lerman et al.[59] use SentiStrength to quantify the sentiment of people in the region of the study.

Authors in [55,116] have studied the sentiment distribution of people discussed in Twitter on politically related events. Kušen et al.[55] have used SentiStrength for sentiment analysis related to the 2016 Austrian presidential elections. Singh et al.[116] have used Meaningcloud API in their study to quantify the sentiment of the people on the issue related to demonetization of 500 and 1000 Indian currency notes. Öztürk and Ayvaz[92] have performed sentiment analysis on social unrest events of the Syria crisis using Twitter data for Turkish and English language. They use RSentiment† tool to classify English tweets while a dictionary-based approach with a manually created lexicon for Turkish tweets.

Authors in [126,73] have studied a comparative evaluation of SA systems in social events through crowdsourcing. Vargas et al.[126] have evaluated the difference of overall sentiment and sentiment expressed in the subject through crowdsourcing of Twitter data related to three crises events. While Maynard et al.[73] have performed a comparative evaluation of SA systems in tweets from social event Earth Hour‡ 2015. They evaluate the difference of annotations via crowdsourcing and a single annotator (one of the author). Further, they evaluate four sentiment analysis tools i.e., SentiStrength, ClimaPinion, GATE-based general domain system, and lexicon-based system over the manually annotated dataset. They observe dif-

*https://en.wikipedia.org/wiki/Hurricane_Sandy
†https://cran.r-project.org/web/packages/RSentiment/index.html
‡https://en.wikipedia.org/wiki/Earth_Hour

ferent tools have different assumptions or domains of interest; thus, they respond differently over the same datasets. The above studies have shown the need for effective sentiment analysis classifier for societal events on social media text to grasp meaningful and valuable insights of the public sentiments. However, as observed in [104,73], these tools are built based on different assumptions and contexts; hence they perform differently on different domains. Therefore, the identification of an appropriate SA tool based on the underlying domain is essential.

4.2.2 Evaluation of Publicly Available Sentiment Analysis Tools

There are limited studies on the evaluation of publicly available sentiment analysis tools. Authors in [104,107,73,1,38] have evaluated some of the available off-the-shelf sentiment analysis tools over various datasets covering the domain of product reviews, movie reviews, social well-being, etc. Ribeiro et al.[104] has evaluated 24 publicly available sentiment analysis tools over non-societal topics such as products, movies, news articles, Youtube videos, etc. While the study of Abbasi et al.[1] have evaluated 20 publicly available sentiment analysis tools using customer reviews discussion on five targeted topics of products and services. SentiStrength, Sentiment140, and Semantria are common among the tools evaluated in these studies[1,104]. It is reported in both studies that most of the publicly available tools respond differently to different datasets of different domains.

While the above two studies focus on product review, movie review, etc. authors in [107,73,38] have considered few selected societal topics to evaluate the publicly available sentiment analysis tools. Saif et al.[107] have considered public opinions on Twitter over topics of election debate* and public health reforms to

*Obama-McCain Debate

evaluate their proposed lexicon-based method Senti-Circle using sentiment lexicon namely MPQA, Thelwall, SentiStrength, and SentiWordnet over the baseline sentiment analysis methods/tools MPQA-Method, SentiStrength, and SentiWordnet. They observe that their method using SentiStrength lexicon is able to outperform the baseline methods. While Maynard et al.[73] have evaluated four publicly available sentiment analysis tools (SentiStength, ClimaPinion*, ARCOMEM, DIVINE) through crowd-sourcing over public opinions on the topics climate change, Earth Hour†, and observe SentiStrength dominating the other three. They also observe that all of the four tools often fail to identify neutral bearing opinions. Gonccalves et al.[38] have evaluated eight publicly available tools over opinions/comments from Youtube videos, MySpace, Twitter, BBC forum. They have further analyzed the agreement of the above tools over various topics, including topics related to airplane crash, elections, sports, and health. They also claim to observe varying responses of the tools over different topics and domains. Unlike the above studies, this study only evaluates publicly available tools in the domains of societal and non-societal topics individually but also evaluates the tools across the domains. Further, we also compare responses of the tools with various locally built sentiment classifiers using state-of-the-art machine learning techniques. Further, it also attempts to find strength and weakness of different tools and methods over different sentiment analysis sub-tasks such as codemixed content, sarcastic comments, comment with aspect/stance, etc.

*http://services.gate.ac.uk/decarbonet/sentiment/api.html
†https://en.wikipedia.org/wiki/Earth_Hour

4.2.3 Evaluation of sentiment analysis techniques

Though there have been a limited number of studies on evaluating the performance of off-the-shelf sentiment analysis tools, a notable number of studies on evaluating sentiment classification techniques and feature engineering methods are reported in the literature. Mostafa et al.[84], and Catal et al.[19] have evaluated various machine learning-based SA techniques in the domain of product reviews. They observed that SVM based classifier outperforms other methods like Naïve Bayes, Decision Tree, k-Nearest Neighbour, etc. They also observed that SA is domain-dependent, and classifiers tend to perform differently on different datasets. Therefore depending on the underlying domain, classifiers need to be retrained. Though this study does not entirely focus on evaluating different sentiment classification techniques, we have also reported the performance of seventeen different classification techniques to compare their performance with that of the off-the-shelf tools. While comparing responses of various `Tools` and `Techniques` over societal and non-societal topics, we report a comparative analysis of different classification techniques as well. In addition to general comparative analysis, we further understand the response of different classification techniques from the perspective of their ability to handle comments with sarcastic nature, stance, code-mixed, etc. Therefore, we briefly discuss some of the existing comparative studies of different sentiment classification techniques using feature engineering methods.

4.3 Experimental Setup

In this study, we have identified ten publicly available SA tools, ten feature-based classification methods, and seven neural network-based classification methods to

evaluate the performances of SA `Tools` and `Techniques`. Tables 4.3 and 4.2 show the list of classifiers we have considered in this study. To assess the classifiers' performances mentioned above, we have considered eight different types of datasets from various domains consisting of societal topics, product reviews, movie reviews, general discussion, etc. In the following sections, we present the experimental setups for evaluating the performances of the classifiers.

4.3.1 Datasets

As mentioned above, we consider eight different types of datasets from various domains to evaluate the performances of the classifiers discussed above. Table 4.1 shows the characteristics and nature of the topics. We utilize `Societal-I`, `SemEval-2013`*, `SemEval-2016`†, and `Sentiment-140`‡ datasets for training as well as testing purposes. While the remaining 4 datasets namely `Societal-II`, `IMDB`, `Amazon`, and `Yelp`§ datasets are considered only for testing the classifiers. The `Societal` datasets (i.e., Societal-I and Societal-II) are locally curated datasets focusing around eight different social issues topics. Except for two topics, the rest of the Societal dataset topics are related to a few events that were trending in India. The two topics that happen outside of India, namely *Syria Crisis*¶, a social unrest event, and *Paris Agreement*‖, an event for climate change, are considered for investigating whether public opinions on social issues are regional dependent. Besides, we have considered the customer review discussion datasets,

*https://www.cs.york.ac.uk/semeval-2013/task2/
†http://saifmohammad.com/WebPages/StanceDataset.htm
‡http://help.sentiment140.com/for-students/
§http://www.yelp.com/dataset_challenge
¶https://en.wikipedia.org/wiki/Syrian_Civil_War
‖https://en.wikipedia.org/wiki/Paris_Agreement

Table 4.1: Characteristics of the Experimental Datasets

Dataset	Pos	Neg	Neu	Total	Topics	Domain
Soceital-I	16375	17047	9000	42422	Kashmir Unrest, Pathankot Attack, Surgical Strike, GSTN*	Social Issue
– Kashmir Unrest	1363	3638	947	5948	–	Social Issue
– Pathankot	1044	3722	1039	5805	–	Social Issue
– Surgical Strike	2116	3278	2191	7585	–	Social Issue
– GSTN	11852	6409	4823	23084	–	Social Issue
Soceital-II	929	2822	705	4456	Demonetization, Uri Attack, Paris Agreement, Syria Crisis	Social Issue
– Demonetization	653	1540	126	2319	–	Social Issue
– Uri Attack	126	416	205	747	–	Social Issue
– Paris Agreement	83	149	147	379	–	Social Issue
– Syria Crisis	67	717	227	1011	–	Social Issue
SemEval-2016	1296	2491	276	4063	Atheism, Climate Change, Feminist Movement, Hillary Clinton, Legalization of Abortion	Social Issue
SemEval-2013	5115	2017	6099	13231	General Discussion	–
Sentiment-140[$]	799978	800024	–	1600002	Consumer reviews discussion	Product Review
IMDB[*]	500	500	–	1000	Movie reviews discussion	Movie Review
Amazon-II[*]	2000000	2000000	–	4000000	Product reviews discussion	Product Review
Yelp[*]	500	500	–	1000	Business reviews discussion	Product Review

[$] Dataset downloaded from http://help.sentiment140.com/for-students/

[*] Dataset downloaded from https://archive.ics.uci.edu/ml/machine-learning-databases/00331/

Table 4.2: Sentiment analysis Tools based on the mode of operation and approach of classification.

SA tools	Methodology	Mode
MeaningCloud	–	Online
SentiStrength	Dictionary based	Offline
IndicoIO	–	Online
Sentiment140	Maximum Entropy	Online
Rsentiment	Dictionary based	Offline
AFINN	Dictionary based	Offline
Pattern.en	Dictionary based	Offline
Emolex	Dictionary based	Offline
SentiWordnet	Dictionary based	Offline
Vader	Dictionary based	Offline

namely Sentiment-140, IMDB, Amazon, and Yelp datasets, for performance comparison of the SA Tools over social issues and customer review discussion domains.

4.3.2 Publicly available sentiment analysis tools

In this section, we discuss in brief the details of the SA tools considered in this study. We have identified 10 SA tools and summarized based on the mode of classifications and approaches in Table 4.2.

- **Meaning Cloud[†]:** It is an online SA tool supporting most European lan-

[†] https://www.meaningcloud.com/developer/sentiment-analysis

guages, namely English, Spanish, French, Italian, etc. It supports the extraction of sentiment at a document or aspect-based level. This tool also has the feature of feeding user-defined dictionaries and models for performing SA. To use this tool, one needs to create an account and generate an API* key. A few limited features with 20,000 sentences per month are allowed for free plan users.

- **SentiStrength†:** It is an offline lexicon-based SA tool. Each word in the lexicon has its corresponding sentiment strength. The lexicon was developed using human-classified MySpace comments. It also supports non-standard spellings and other conventional textual methods of expressing sentiment. We can download this tool for offline usage by registering through the tool website.

- **IndicoIO‡:** It is an online SA tool that deals with the English language. This tool returns the sentiment score of the input text in the range between 0 to 1. Therefore, we define the sentiment score higher than 0.6 is treated as positive, while the score lesser than 0.4 is treated as negative sentiment and in between as the neutral sentiment for three-class sentiment classification. While in two-class SA, the sentiment score higher than 0.5 is treated as positive; otherwise, it is treated as negative. To use this tool, one needs to create an account and generate an API key. A few limited features with 10,000 sentences per month are allowed for free plan users.

*Application Programming Interface
†http://sentistrength.wlv.ac.uk
‡https://indico.io

- **Sentiment-140**[*]**:** It is an online SA tool build using product review discussion in Twitter supporting English and Spanish languages. This tool does not require a user account and has no restriction on the number of text samples. The dataset used for building this tool has been open source for academic purposes. We consider this dataset for evaluating the SA `Tools` and `Techniques` considered in this study.

- **RSentiment**[†]**:** It is an offline R programming language based SA tool package that deals with the English language. It detects the input text's sentiment through the task of natural language processing such as Parts of Speech tagging, Stemming, etc. as well as detecting sarcasm, negations, and various degrees of adjectives and emoticons. It can detect sarcasm based on the usage of punctuation marks.

- **AFINN**[‡]**:** It is an offline lexicon-based SA tool access through Python programming language package. The lexicon used in this tool is of English words manually annotated based on the words' emotion intensity.

- **Pattern.en**[§]**:** It is an offline lexicon-based SA tool accessed through a Python programming language package. It detects the input text's sentiment through the task of natural language processing such as Part-of-Speech tagging, Stemming, etc., and calculates the polarity of the adjectives and adverbs from the lexicon.

- **Emolex**[¶]**:** We consider the Emotion Lexicon (Emolex) and build an in-house

[*] http://help.sentiment140.com/for-students
[†] https://www.rdocumentation.org/packages/RSentiment/versions/2.2.2
[‡] http://corpustext.com/reference/sentiment_afinn.html
[§] https://www.clips.uantwerpen.be/pages/pattern-en
[¶] https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm

Table 4.3: List of classifiers

Notation	Feature-based Classifiers	Notation	Neural network-based classifiers
DT	Decision Tree	MLP	Multi Layer Perceptron
kNN	k-Nearest Neighbour	CNN	Convolution Neural Network
SVM (NL)	SVM Non-Linear	D-CNN	Deep CNN
SVM	SVM Linear	RNN	Recurrent Neural Network
LR	Logistic Regression	LSTM	Long Short Term Memory
RF	Random Forest	Bi-LSTM	Bidirectional LSTM
A-Boost (R)	AdaBoost (Real)	CNN-BiLSTM	CNN + Bi-LSTM
A-Boost (D)	AdaBoost (Discrete)		
ET	Extra Trees		
GB	Gradient Boosting		

lexicon-based SA classifier. This classifier is similar to the approach of RSentiment and Pattern.en tools except for the sentiment lexicon.

- **SentiWordnet*:** We build an in-house lexicon-based SA classifier using the SentiWordNet lexicon. Each WordNet synset from the SentiWordNet lexicon has three quantitative scores describing positive, negative, and neutral sentiments for each term in the synsets.

- **Vader†:** It is an offline lexicon-based SA tool access through Python programming language package. This tool provides the confidence score of positive or negative sentiment classified. It can also handle social media text written in the English language.

4.3.3 Sentiment analysis techniques

We consider a total of 17 machine learning techniques; consisting of 10 feature-based classifiers and 7 neural network-based classifiers to evaluate the performances of SA `Techniques`. The details of these classifiers are shown in Table 4.3. The feature-based classifiers are build using Scikit-learn‡ machine learning toolkit

*http://sentiwordnet.isti.cnr.it/
†https://github.com/cjhutto/vaderSentiment
‡http://scikit-learn.org/stable/index.html

while the neural network-based classifiers are build using Keras* deep learning Python libraries.

Before considering the `Societal` dataset for various experiments, the dataset is pre-processed to remove stopwords, embedded URL, twitter specific keywords like RT. Similar to the studies in [135,53,93,26,95] unigrams, hashtags, and emoticons features are considered for feature-based classifiers. Considering the large number of features (many of which may not be useful for classification), the distinctive characteristics of the features are estimated using Entropy[110] and Pointwise Mutual Information (PMI)[24]. The entropy of each feature is estimated across sentiment classes to measure information content in the features, while the PMI between features and sentiment classes measures the Strength of Association (SOA) of the features across the sentiment classes. The candidate features with low entropy score (less than 0.3) and a high SOA score (greater than average SOA score of each class label) are selected as the final features. In addition to these features, the well-known emoticons† and emojis‡ used in [53,93] are also included. We build the explicit feature-based classifiers using the Scikit-learn packages with default parameters.

To avoid explicit feature engineering as discussed above, we consider seven neural network-based classifiers used in the studies[68,88]. For all these classifiers, we use word embedding (low dimensional vector) via SkipGram model[77]. We represent the tweets into matrices using the word embedding vectors for words present in the tweets. For unifying the tweet matrix's size, we hypothesize the length of a tweet to 40 words. We padded zero vectors for those tweets that have

*https://keras.io/
†https://en.wikipedia.org/wiki/List_of_emoticons
‡http://kt.ijs.si/data/Emoji_sentiment_ranking/

word length less than 40. For the CNN classifier, we use 128 filters of kernel size 3 and rectified linear function (ReLu) as the activation function. While in Deep CNN, we add two hidden layers (a combination of convolution and max-pooling in each layer) using the same CNN classifier's parameters. Similar to the study of Lu et al.[68], we define the embedding size of the hidden layer for RNN, LSTM, and Bi-LSTM architectures to be 100. For CNN+BiLSTM configuration, we first apply convolution on the input layer and then pass the filter outputs to BiLSTM architecture. We use a softmax activation function in the output layer for each DNN classifier and categorical cross-entropy loss function for estimating the loss. Considering the hyperparameters mentioned above, we train the neural network-based SA classifiers.

4.3.4 Evaluation metrics

We consider the traditional evaluation metrics for classification, such as Accuracy, Precision, Recall, and F1 scores, to evaluate the performance of the SA classifiers. We calculate the FMacro score for each classifier by taking average F1 scores over the three sentiment classes. In this study, we consider the Accuracy (Acc) and the FMacro (FM) scores to evaluate the performances of SA `Tools` and `Techniques`.

4.4 Results and observations

4.4.1 Performance of publicly available sentiment analysis tools

To answer the question *"Are the off-the-shelf sentiment analysis tools suitable for finding public sentiment over societal topics?"*, this section investigates the performance of 10 sentiment analysis `Tools` on different types of datasets of different

Table 4.4: Performance of sentiment analysis tools in different types of testing datasets

List of Classifiers	Societal domain											
	Societal-I		Uri Attack		Demonetization		Syria Crisis		Paris		Societal-II	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MeaningCloud	0.60	0.59	0.58	0.34	0.58	0.49	0.59	0.42	0.75	0.74	0.63	0.60
SentiStrength	0.55	0.54	0.60	0.32	0.53	0.43	0.69	0.42	0.80	0.79	0.49	0.47
IndicoIO	0.44	0.42	0.34	0.53	0.38	0.36	0.49	0.42	0.60	0.48	0.49	0.49
Sentiment140	0.35	0.34	0.38	0.49	0.24	0.24	0.29	0.27	0.40	0.36	0.36	0.35
Rsentiment	0.57	0.55	0.52	0.55	0.52	0.43	0.60	0.53	0.64	0.59	0.55	0.50
AFINN	0.63	0.59	0.62	0.52	**0.59**	**0.50**	0.61	0.46	**0.86**	**0.84**	**0.65**	**0.62**
Pattern.en	0.40	0.40	0.39	0.38	0.35	0.33	0.34	0.31	0.58	0.53	0.40	0.40
Emolex	0.36	0.40	0.29	0.29	0.20	0.36	0.29	0.23	0.40	0.30	0.27	0.27
SentiWordnet	0.22	0.22	0.30	0.28	0.08	0.06	0.25	0.15	0.58	0.26	0.25	0.15
Vader	**0.63**	**0.62**	**0.62**	**0.57**	0.58	0.49	**0.72**	**0.51**	0.75	0.61	0.64	0.59
Average	0.48	0.47	0.46	0.43	0.41	0.37	0.49	0.37	0.63	0.55	0.47	0.44

List of Classifiers	SemEval Challenge				Customer review domain							
	SemEval-16		SemEval-13		Sentiment-140		IMDB		Amazon		Yelp	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MeaningCloud	0.53	0.47	0.62	0.60	**0.73**	**0.73**	0.85	0.85	0.89	0.89	0.88	0.87
SentiStrength	**0.58**	**0.56**	0.61	0.59	0.64	0.63	0.75	0.74	0.79	0.79	0.80	0.80
IndicoIO	0.44	0.39	0.57	0.54	0.66	0.65	**0.95**	**0.95**	**0.95**	**0.95**	**0.93**	**0.93**
Sentiment140	0.24	0.26	0.27	0.26	0.51	0.51	0.57	0.57	0.48	0.44	0.56	0.56
Rsentiment	0.51	0.44	0.51	0.48	0.44	0.51	0.82	0.81	0.86	0.85	0.84	0.83
AFINN	0.53	0.47	0.64	0.61	0.70	0.70	0.77	0.77	0.80	0.80	0.77	0.77
Pattern.en	0.34	0.33	0.57	0.53	0.69	0.68	0.74	0.73	0.69	0.66	0.70	0.68
Emolex	0.24	0.42	0.47	0.36	0.54	0.45	0.61	0.59	0.59	0.56	0.59	0.57
SentiWordnet	0.08	0.03	0.39	0.34	0.71	0.70	0.70	0.69	0.74	0.74	0.70	0.69
Vader	0.52	0.46	**0.64**	**0.62**	0.71	0.70	0.79	0.79	0.84	0.83	0.81	0.81
Average	0.40	0.38	0.53	0.49	0.63	0.63	0.75	0.75	0.76	0.75	0.76	0.75

domains to identify which `Tool` performs better than its counterparts. We further investigate the performance of each `Tool` to identify the domain on which they perform the best. Table 4.4 shows the performances of these `Tools` evaluated over various domain datasets. On comparing the performances of each `Tool` for different type of datasets (i.e. column-wise comparison in the Table 4.4), we observed that Vader dominates other `Tools` on four datasets namely Societal-I, SemEval-2013, Uri Attack, and Syria Crisis. While IndicoIO dominates other `Tools` on customer review datasets, i.e., IMDB, Amazon, and Yelp datasets. AFINN dominates the Societal-II dataset and, more specifically, on its two topics, i.e., Demonetization and Paris Agreement. MeaningCloud and SentiStrength dominate on Sentiment-140 and SemEval-2016 datasets, respectively. Further, on evaluat-

ing the performance of each `Tool` on different types of datasets (i.e., row-wise comparison in the Table 4.4), we observed that most of the SA `Tools` have performed better on the customer review such as Amazon, IMDB, and Yelp datasets as compared to societal and generic datasets. Interestingly, AFINN has shown better performance on the topic Paris Agreement than over the customer review datasets. Low performance of these `Tools` over societal topics may be due to various factors such as noisy texts, presence of phonetically typed non-English words, code-mixed, etc. It is evident from Table 4.4 that, on average, the performance of the `Tools` on customer review dominates societal topics and general discussion.

We further perform dominance tests of different `Tools` and `Techniques` across different experimental setups. Let X and Y be two sets of experimental results. Say, for example, X is the set of performances of MeaningCloud over societal topics and Y be the set of performances of MeaningCloud over non-societal topics. The dominance score of the set X over Y is defined as the likelihood of a randomly picked up instance of classifier's performance in X outperforms another randomly picked up instance in Y. If n be the number of (x, y) pairs such that $x > y, x \in X, y \in Y$, the dominance score of X over Y is defined as below.

$$Dominance(X > Y) = \frac{100n}{|X|.|Y|} \tag{4.1}$$

In Figures 4.1 and 4.2, we summarized the scalability test of the `Tools` reported in Table 4.2 across societal and customer review domains, and across English and code-mixed datasets, respectively. Figure 4.1 clearly shows that all of the publicly available tools respond better on customer reviews domain than their societal counterparts. Interestingly there is not even a single instance for *IndicolO, Sen-*

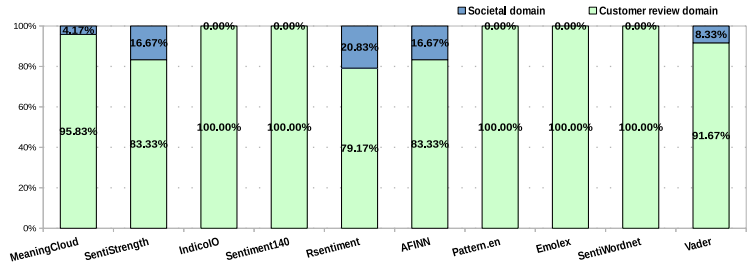


Figure 4.1: Dominance test of sentiment analysis Tools over Societal domains vs Customer review domains

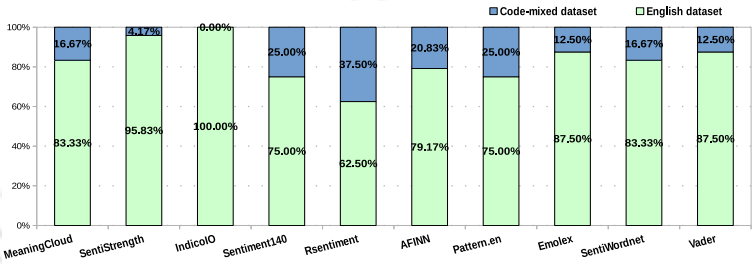


Figure 4.2: Dominance test sentiment analysis Tools over Code-mixed text vs English language text

timent140, Pattern.en, Emolex and *SentiWordnet* that they dominate on societal topics. Among these tools, *RSentiment* dominates others with 20.83% on societal topics, and *SentiStrength* and *AFINN* with 16.67%. Therefore, it is evident from the observations in Table 4.4 and Figure 4.1 that off-the-shelf tools are biased towards customer reviews, and not suitable for the societal domain.

Further, in Figure 4.2, we investigate the responses of these tools on code-mixed and non-code-mixed datasets. The `Societal-I`, `Demonetization` and `Uri Attack` are considered to be code-mixed as it contains phonetically typed words in Hindi. The remaining testing datasets, including the Paris Agreement and Syria Crisis, are considered non-code-mixed as written in the English language. It is evident from the figure that all of these tools are suitable for English language. However, among them *RSentiment* is able to handle code-mixed text better than its counterparts, and then *Sentiment140* and *Pattern.en* follows.

4.4.2 Performance of sentiment analysis techniques

To understand if there is a need to build sentiment classifier instead of using off-the-shelf tools for analyzing sentiment of public opinion on societal topics, we further build 17 classifiers using state-of-the-art machine learning techniques (listed in Table 4.3) over the datasets shown in Table 4.1. These classifiers are further evaluated over various homogeneous and heterogeneous setups. For homogeneous setup, both the train and test samples are taken from the same dataset. While for heterogeneous setup, the train and test samples are taken from different datasets.

Tables 4.5, 4.6, 4.7 and 4.8 shows the performance of locally built sentiment classifiers over various datasets Societal-I, SemEval-2013, SemEval-2016 and Sentiment-140 datasets respectively. For the Societal-I and the Sentiment-140 datasets, we use 10-fold cross-validations. Whereas, for SemEval datasets, we consider the train and test samples provided with the datasets. The boldface entries (in blue color) show the performance of different classifiers in a homogeneous setup (i.e., trained and tested on the same dataset). It clearly shows that CNN based classifiers dominate other classifiers in most of the datasets. On average, the neural network-based classifiers dominate the feature-based classifiers in the majority of the cases in homogeneous setups (in 3 out of 4 datasets, namely Societal-I, SemEval-2016, and Sentiment-140).

To evaluate the response of the Techniques across domains, we further investigate the performance of the locally built classifiers in heterogeneous setups (cross-domain analysis). Tables 4.5, 4.6, 4.7 and 4.8 shows the performance of individual classifier over different test datasets in heterogeneous setups. Figure 4.3 summarize the cross-domain performance of different classifiers reported

Table 4.5: Performance of classifiers trained with Societal-I datasets. The boldfaces represent the classifiers outperforming other classifiers over various testing dataset. The boldfaces in blue color shows the best performing classifiers in homogeneous setup i.e. trained and tested on same dataset.

	Societal domain											
List of Classifiers	Societal-I		Uri Attack		Demonetization		Syria Crisis		Paris		Societal-II	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MLP	0.72	0.70	0.51	0.46	0.61	0.51	0.66	0.42	0.51	0.41	0.56	0.42
CNN	**0.76**	**0.74**	0.66	0.46	0.63	0.50	**0.79**	0.51	**0.71**	**0.69**	0.62	0.51
D-CNN	**0.76**	**0.74**	0.64	0.49	0.59	0.47	0.70	0.51	**0.71**	**0.69**	0.56	0.48
RNN	0.74	0.72	0.54	0.39	0.53	0.47	0.65	0.47	0.56	0.54	0.53	0.46
LSTM	0.75	0.74	0.61	0.43	0.59	0.48	0.76	**0.54**	0.63	0.61	0.55	0.48
Bi-LSTM	0.75	0.74	0.55	0.38	0.63	0.53	0.61	0.37	0.63	0.61	0.58	0.48
CNN-BiLSTM	0.75	0.74	0.61	0.42	0.63	0.52	0.77	0.49	0.67	0.64	0.58	0.46
Average	0.75	0.73	0.59	0.43	0.60	0.54	0.71	0.47	0.63	0.60	0.57	0.47
DT	0.73	0.72	0.63	0.56	0.55	0.53	0.65	0.43	0.49	0.48	0.56	0.48
kNN	0.66	0.65	0.64	0.49	0.53	0.55	0.72	0.42	0.59	0.57	0.64	0.52
SVM (NL)	0.46	0.39	0.32	0.31	0.34	0.32	0.25	0.28	0.52	0.42	0.35	0.30
SVM	**0.77**	**0.74**	0.66	0.60	0.60	0.57	0.71	0.50	0.71	0.68	**0.66**	**0.61**
LR	0.76	0.74	0.53	0.45	**0.70**	**0.58**	0.74	0.38	0.19	0.13	0.63	0.41
RF	0.75	0.73	**0.75**	**0.72**	0.28	0.22	0.07	0.04	0.44	0.20	0.18	0.10
A-Boost (R)	0.73	0.70	0.72	0.71	0.05	0.05	0.21	0.11	0.38	0.19	0.18	0.10
A-Boost (D)	0.63	0.51	0.63	0.52	0.67	0.55	0.72	0.28	0.17	0.10	0.56	0.24
ET	0.74	0.73	0.74	0.72	0.66	0.55	0.69	0.29	0.17	0.11	0.58	0.31
GB	0.68	0.62	0.68	0.63	0.67	0.54	0.72	0.28	0.17	0.10	0.56	0.24
Average	0.69	0.65	0.63	0.57	0.51	0.57	0.55	0.30	0.38	0.30	0.49	0.33

	SemEval Challenge				Customer review domain							
List of Classifiers	SemEval-16		SemEval-13		Sentiment-140		IMDB		Amazon		Yelp	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MLP	0.54	0.38	0.37	0.30	0.59	0.53	0.62	0.57	0.69	0.67	0.63	0.61
CNN	0.55	0.31	0.46	0.45	0.47	0.39	0.65	0.65	**0.72**	**0.72**	0.67	0.67
D-CNN	0.49	0.35	0.46	0.45	0.47	0.39	**0.69**	**0.69**	0.72	0.71	0.68	0.67
RNN	0.51	0.37	0.42	0.41	0.47	0.42	0.63	0.63	0.67	0.67	0.64	0.63
LSTM	0.52	0.34	0.45	0.44	0.47	0.40	0.64	0.64	0.71	0.71	0.67	0.66
Bi-LSTM	0.55	0.41	0.43	0.42	0.47	0.38	0.64	0.64	0.72	0.71	0.67	0.66
CNN-BiLSTM	0.58	0.35	0.42	0.40	0.47	0.39	0.65	0.64	0.72	0.71	0.69	0.69
Average	0.53	0.36	0.43	0.41	0.49	0.41	0.65	0.64	0.71	0.70	0.66	0.66
DT	0.57	0.48	0.47	0.46	0.62	0.62	0.64	0.64	0.69	0.69	0.68	0.68
kNN	0.65	0.46	0.49	0.47	0.59	0.56	0.65	0.64	0.66	0.63	0.63	0.59
SVM (NL)	0.34	0.27	0.40	0.31	0.54	0.46	0.50	0.41	0.52	0.41	0.55	0.44
SVM	0.61	**0.49**	**0.52**	**0.51**	**0.64**	**0.64**	0.68	0.68	0.70	0.70	**0.70**	**0.70**
LR	0.68	0.36	0.21	0.17	0.56	0.44	0.56	0.45	0.63	0.56	0.57	0.49
RF	0.25	0.13	0.42	0.20	0.49	0.33	0.49	0.33	0.49	0.33	0.51	0.34
A-Boost (R)	0.07	0.04	0.41	0.20	0.49	0.33	0.49	0.33	0.49	0.33	0.51	0.34
A-Boost (D)	**0.70**	0.27	0.16	0.09	0.51	0.34	0.51	0.34	0.51	0.34	0.49	0.33
ET	0.67	0.32	0.42	0.20	0.49	0.35	0.49	0.35	0.50	0.37	0.52	0.36
GB	0.67	0.27	0.16	0.09	0.51	0.34	0.51	0.34	0.51	0.34	0.49	0.33
Average	0.52	0.31	0.37	0.27	0.54	0.44	0.55	0.45	0.57	0.47	0.57	0.46

in the above tables by showing the number of dominating test cases in each test dataset for each classifier built over different training datasets. Among the classifiers built over `Societal-I` dataset, SVM dominates other classifiers in five test datasets. CNN and LSTM built over the `SemEval-2013` dominates in three

Table 4.6: Performance of classifiers trained with SemEval 2013 datasets. The boldfaces represent the classifiers outperforming other classifiers over various testing dataset. The boldfaces in blue color shows the best performing classifiers in homogeneous setup i.e. trained and tested on same dataset.

	Societal domain											
List of Classifiers	Societal-I		Uri Attack		Demonetization		Syria Crisis		Paris		Societal-II	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MLP	0.40	0.23	0.19	0.13	0.27	0.18	0.08	0.07	0.44	0.22	0.26	0.16
CNN	0.32	0.30	0.43	0.40	0.42	0.35	**0.50**	**0.42**	0.49	0.46	**0.45**	**0.44**
D-CNN	0.30	0.28	0.46	**0.43**	0.38	0.33	0.34	0.30	0.49	0.46	0.41	0.40
RNN	0.31	0.29	0.31	0.31	0.33	0.30	0.17	0.17	0.20	0.20	0.27	0.26
LSTM	0.31	0.28	**0.47**	0.41	**0.45**	**0.36**	0.32	0.30	0.39	0.37	0.42	0.38
Bi-LSTM	0.31	0.29	0.43	0.40	0.41	0.34	0.39	0.35	0.32	0.34	0.39	0.37
CNN-BiLSTM	0.31	0.29	0.41	0.36	0.35	0.30	0.27	0.26	0.39	0.37	0.35	0.34
Average	0.32	0.28	0.38	0.35	0.38	0.31	0.30	0.27	0.39	0.35	0.36	0.33
DT	0.39	0.39	0.39	0.38	0.29	0.28	0.32	0.29	**0.65**	**0.56**	0.39	0.39
kNN	0.22	0.16	0.28	0.19	0.09	0.08	0.24	0.14	0.40	0.21	0.21	0.15
SVM (NL)	0.38	0.33	0.29	0.29	0.34	0.29	0.26	0.26	0.42	0.36	0.33	0.34
SVM	**0.40**	**0.40**	0.44	0.42	0.28	0.28	0.44	0.37	0.47	0.37	0.38	0.37
LR	0.26	0.21	0.29	0.23	0.08	0.08	0.23	0.21	0.41	0.22	0.21	0.16
RF	0.20	0.11	0.26	0.14	0.05	0.03	0.22	0.12	0.39	0.19	0.19	0.11
A-Boost (R)	0.20	0.11	0.26	0.14	0.05	0.03	0.22	0.12	0.39	0.19	0.19	0.11
A-Boost (D)	0.20	0.11	0.26	0.14	0.05	0.03	0.22	0.12	0.39	0.19	0.19	0.11
ET	0.39	0.20	0.17	0.12	0.28	0.15	0.07	0.06	0.63	0.45	0.30	0.26
GB	0.20	0.11	0.26	0.14	0.05	0.03	0.22	0.12	0.39	0.19	0.19	0.11
Average	0.28	0.21	0.29	0.22	0.16	0.13	0.25	0.18	0.45	0.29	0.26	0.21

	SemEval Challenge				Customer review domain							
List of Classifiers	SemEval-16		SemEval-13		Sentiment-140		IMDB		Amazon		Yelp	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MLP	0.30	0.17	0.57	0.52	0.51	0.34	0.51	0.34	0.51	0.34	0.50	0.34
CNN	0.43	0.38	0.61	0.57	**0.64**	**0.64**	0.69	0.68	**0.72**	**0.71**	0.69	0.68
D-CNN	0.43	0.39	**0.62**	**0.57**	0.62	0.62	**0.71**	**0.71**	0.70	0.70	**0.71**	**0.71**
RNN	0.33	0.28	0.55	0.49	0.59	0.58	0.63	0.62	0.61	0.59	0.63	0.62
LSTM	**0.46**	**0.40**	0.58	0.54	0.62	0.62	0.68	0.68	0.70	0.69	0.68	0.68
Bi-LSTM	0.38	0.36	0.59	0.54	0.63	0.62	0.67	0.65	0.67	0.65	0.67	0.65
CNN-BiLSTM	0.37	0.35	0.58	0.54	0.63	0.63	0.70	0.69	0.71	0.71	0.70	0.69
Average	0.39	0.33	0.58	0.54	0.61	0.58	0.65	0.62	0.66	0.63	0.65	0.62
DT	0.33	0.34	0.61	0.54	0.46	0.35	0.66	0.64	0.62	0.58	0.60	0.57
kNN	0.11	0.10	0.46	0.27	0.48	0.34	0.57	0.51	0.56	0.49	0.55	0.47
SVM (NL)	0.37	0.32	0.43	0.40	0.44	0.34	0.57	0.52	0.57	0.50	0.56	0.48
SVM	0.35	0.33	0.61	0.55	0.42	0.35	0.69	0.68	0.69	0.68	0.68	0.67
LR	0.16	0.19	0.57	0.53	0.51	0.34	0.51	0.34	0.51	0.34	0.50	0.34
RF	0.07	0.04	**0.65**	0.55	0.51	0.34	0.51	0.34	0.51	0.34	0.50	0.34
A-Boost (R)	0.07	0.04	0.61	0.52	0.51	0.34	0.51	0.34	0.51	0.34	0.50	0.34
A-Boost (D)	0.07	0.04	0.61	0.47	0.51	0.34	0.51	0.34	0.51	0.34	0.50	0.34
ET	0.26	0.16	0.64	**0.56**	0.48	0.34	0.52	0.40	0.52	0.39	0.51	0.39
GB	0.07	0.04	0.64	**0.56**	0.51	0.34	0.51	0.34	0.51	0.34	0.50	0.34
Average	0.19	0.16	0.58	0.50	0.48	0.34	0.56	0.45	0.55	0.43	0.54	0.44

datasets each. CNN built over the `SemEval-2016` dominates in five and RNN dominates for `Sentiment-140` in six datasets. While counting the number of dominating cases across all four training datasets, CNN dominates twelve test cases. Logistic Regression (LR) classifier stands next with eight dominating test

Table 4.7: Performance of classifiers trained with SemEval 2016 datasets. The boldfaces represent the classifiers outperforming other classifiers over various testing dataset. The boldfaces in blue color shows the best performing classifiers in homogeneous setup i.e. trained and tested on same dataset.

	Societal domain											
List of Classifiers	Societal-I		Uri Attack		Demonetization		Syria Crisis		Paris		Societal-II	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MLP	0.42	0.25	0.43	0.24	0.63	0.29	0.59	0.30	0.41	0.29	0.56	0.29
CNN	0.48	0.40	0.51	0.39	0.58	0.39	**0.70**	**0.41**	0.46	0.43	0.58	0.42
D-CNN	0.44	0.32	0.29	0.24	0.42	0.30	0.50	0.30	0.35	0.33	0.41	0.31
RNN	0.41	0.36	0.41	0.33	0.50	0.34	0.44	0.32	0.36	0.33	0.46	0.35
LSTM	0.48	**0.43**	0.52	**0.42**	0.56	0.39	0.66	0.39	0.45	0.44	0.57	0.42
Bi-LSTM	**0.49**	0.40	**0.54**	0.40	**0.60**	**0.41**	0.68	0.37	**0.46**	**0.44**	**0.60**	**0.42**
CNN-BiLSTM	0.42	0.40	0.48	0.40	0.45	0.33	0.65	**0.43**	0.45	0.43	0.50	0.39
Average	0.45	0.37	0.45	0.34	0.53	0.35	0.60	0.36	0.42	0.38	0.53	0.37
DT	0.45	0.36	0.41	0.34	0.54	0.35	0.48	0.33	0.42	0.35	0.49	0.36
kNN	0.48	0.35	0.27	0.22	0.49	0.33	0.37	0.25	0.39	0.32	0.41	0.29
SVM (NL)	0.42	0.27	0.17	0.14	0.30	0.20	0.13	0.11	0.32	0.31	0.24	0.20
SVM	0.48	0.38	0.37	0.30	0.54	0.36	0.41	0.27	0.37	0.31	0.46	0.34
LR	0.42	0.20	0.43	0.20	**0.67**	0.27	**0.71**	0.28	0.39	0.19	0.61	0.25
RF	0.41	0.22	0.43	0.25	0.66	0.29	0.69	0.33	0.40	0.23	0.60	0.29
A-Boost (R)	0.41	0.22	0.43	0.25	0.66	0.29	0.69	0.33	0.40	0.23	0.60	0.29
A-Boost (D)	0.42	0.20	0.43	0.20	**0.67**	0.27	**0.71**	0.28	0.39	0.19	0.61	0.25
ET	0.42	0.20	0.43	0.20	**0.67**	0.27	**0.71**	0.28	0.39	0.19	0.61	0.25
GB	0.42	0.20	0.43	0.20	**0.67**	0.27	**0.71**	0.28	0.39	0.19	0.61	0.25
Average	0.43	0.26	0.38	0.23	0.59	0.29	0.56	0.27	0.39	0.25	0.52	0.28

	SemEval Challenge				Customer review domain							
List of Classifiers	SemEval-16		SemEval-13		Sentiment-140		IMDB		Amazon		Yelp	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MLP	0.63	0.47	0.22	0.19	0.06	0.10	0.51	0.45	0.51	0.39	0.51	0.41
CNN	**0.69**	0.47	0.39	0.35	**0.60**	**0.60**	**0.62**	**0.62**	0.62	0.62	**0.65**	**0.65**
D-CNN	0.64	0.48	**0.42**	0.30	0.57	0.55	0.56	0.54	0.58	0.54	0.57	0.53
RNN	0.62	0.40	0.33	0.31	0.51	0.45	0.55	0.51	0.54	0.49	0.56	0.52
LSTM	0.66	**0.50**	0.40	0.34	0.57	0.57	**0.62**	**0.62**	0.64	0.64	0.64	0.64
Bi-LSTM	**0.69**	0.47	0.39	0.33	0.57	0.57	0.60	0.60	0.64	0.64	0.61	0.61
CNN-BiLSTM	0.63	0.48	0.41	**0.39**	0.59	0.59	**0.62**	**0.62**	**0.65**	**0.65**	0.63	0.62
Average	0.65	0.47	0.36	0.32	0.50	0.49	0.58	0.57	0.60	0.57	0.60	0.57
DT	0.57	0.41	0.36	0.32	0.25	0.25	0.53	0.53	0.63	0.63	0.53	0.53
kNN	0.59	0.40	0.36	0.28	0.41	0.32	0.54	0.48	0.53	0.48	0.53	0.49
SVM (NL)	0.38	0.29	0.40	0.29	0.41	0.32	0.52	0.44	0.49	0.43	0.50	0.44
SVM	0.66	**0.45**	0.36	0.31	0.23	0.26	0.56	0.56	0.60	0.60	0.59	0.59
LR	0.61	0.44	0.16	0.09	0.00	0.00	0.47	0.32	0.51	0.34	0.51	0.34
RF	0.63	0.44	0.21	0.18	0.16	0.18	0.49	0.47	0.51	0.50	0.45	0.43
A-Boost (R)	0.62	0.43	0.21	0.18	0.16	0.18	0.49	0.47	0.51	0.50	0.45	0.43
A-Boost (D)	**0.68**	0.40	0.16	0.09	0.00	0.00	0.47	0.32	0.51	0.34	0.51	0.34
ET	0.62	0.43	0.16	0.09	0.03	0.05	0.49	0.41	0.53	0.43	0.53	0.42
GB	**0.68**	0.42	0.16	0.09	0.00	0.00	0.47	0.32	0.51	0.34	0.51	0.34
Average	0.60	0.41	0.25	0.19	0.16	0.16	0.50	0.43	0.53	0.46	0.51	0.43

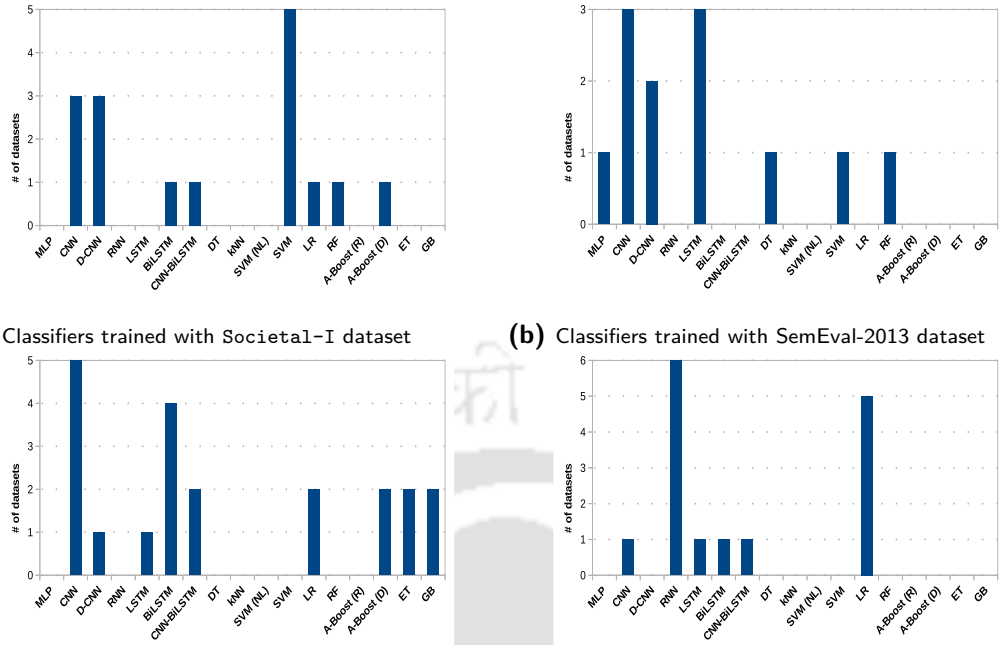
cases. Among the classifiers, k-Nearest Neighbour, SVM (Non-linear kernel), and AdaBoost (Real) have performed the least. Further, we perform the scalability test between feature-based and neural network-based classifiers over the results reported in Tables 4.5, 4.6, 4.7, and 4.8 through the dominance test shown in

Table 4.8: Performance of 2-class classifiers trained with Sentiment-140 datasets. The boldfaces represent the classifiers outperforming other classifiers over various testing dataset. The boldfaces in blue color shows the best performing classifiers in homogeneous setup i.e. trained and tested on same dataset.

	Societal domain											
List of Classifiers	Societal-I		Uri Attack		Demonetization		Syria Crisis		Paris		Societal-II	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MLP	0.51	0.46	0.35	0.35	0.45	0.45	0.19	0.19	0.62	0.54	0.38	0.36
CNN	0.62	0.62	0.61	0.57	0.59	0.56	0.62	0.49	0.75	0.70	0.60	0.56
D-CNN	0.59	0.59	0.58	0.54	0.56	0.54	0.66	0.51	0.82	0.78	0.59	0.55
RNN	0.60	0.61	0.59	0.55	0.64	**0.59**	0.59	0.48	**0.88**	**0.86**	0.63	0.58
LSTM	0.60	0.61	0.57	0.53	0.58	0.56	0.50	0.41	0.44	0.44	0.57	0.53
Bi-LSTM	0.61	0.61	0.57	0.53	0.59	0.56	0.65	0.51	0.66	0.64	0.60	0.56
CNN-BiLSTM	0.59	0.59	0.68	0.51	0.60	0.57	**0.79**	0.51	0.75	0.71	0.64	0.57
Average	0.59	0.58	0.57	0.51	0.57	0.55	0.57	0.44	0.70	0.66	0.57	0.53
DT	0.61	0.61	0.61	0.56	0.60	0.55	0.72	0.53	0.55	0.54	0.61	0.52
SVM (NL)	0.58	0.57	0.33	0.32	0.38	0.36	0.30	0.29	0.74	0.53	0.38	0.34
SVM	0.53	0.48	0.61	**0.59**	0.58	0.56	0.60	0.50	0.72	0.68	0.63	**0.59**
LR	**0.63**	**0.62**	**0.78**	0.51	**0.71**	0.48	0.72	**0.63**	0.41	0.40	**0.69**	0.50
RF	0.55	0.44	0.24	0.19	0.30	0.23	0.09	0.08	0.71	0.42	0.45	0.23
A-Boost (R)	0.49	0.33	0.24	0.19	0.30	0.23	0.09	0.08	0.71	0.42	0.45	0.23
A-Boost (D)	0.49	0.33	0.24	0.19	0.30	0.23	0.09	0.08	0.71	0.42	0.45	0.23
ET	0.49	0.33	0.72	0.51	0.67	0.45	0.86	0.55	0.27	0.22	0.63	0.43
GB	0.51	0.42	0.24	0.19	0.30	0.23	0.09	0.08	0.71	0.42	0.45	0.23
Average	0.54	0.46	0.44	0.36	0.46	0.37	0.42	0.31	0.62	0.45	0.48	0.33

	SemEval Challenge				Customer review domain							
List of Classifiers	SemEval-16		SemEval-13		Sentiment-140		IMDB		Amazon		Yelp	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MLP	0.39	0.38	0.73	0.61	0.69	0.69	0.57	0.50	0.59	0.54	0.55	0.46
CNN	0.60	**0.59**	0.67	0.64	0.76	0.76	0.72	0.72	0.72	0.71	0.72	0.72
D-CNN	0.55	0.54	0.69	0.66	0.75	0.75	0.74	0.74	0.73	0.73	0.74	0.74
RNN	0.58	0.58	**0.75**	**0.71**	0.76	0.76	**0.77**	**0.76**	**0.76**	**0.76**	**0.77**	**0.76**
LSTM	0.56	0.55	0.70	0.66	0.76	0.76	0.75	0.75	0.73	0.73	0.75	0.75
Bi-LSTM	0.58	0.57	0.71	0.68	**0.76**	**0.77**	0.73	0.73	0.74	0.74	0.73	0.73
CNN-BiLSTM	0.61	**0.59**	0.68	0.65	**0.76**	**0.77**	0.73	0.73	0.71	0.71	0.72	0.72
Average	0.55	0.54	0.70	0.66	0.75	0.75	0.72	0.71	0.71	0.70	0.71	0.70
DT	0.57	0.56	0.65	0.62	0.69	0.69	0.63	0.63	0.68	0.68	0.64	0.64
SVM (NL)	0.41	0.40	0.62	0.54	0.68	0.68	0.53	0.49	0.57	0.51	0.55	0.49
SVM	0.54	0.53	0.69	0.65	0.59	0.55	0.71	0.71	0.72	0.72	0.71	0.71
LR	**0.71**	0.54	0.42	0.42	**0.74**	**0.74**	0.55	0.45	0.57	0.48	0.58	0.51
RF	0.32	0.24	0.73	0.42	0.73	0.73	0.50	0.33	0.50	0.34	0.50	0.33
A-Boost (R)	0.32	0.24	0.73	0.42	0.72	0.72	0.50	0.33	0.50	0.34	0.50	0.33
A-Boost (D)	0.32	0.24	0.73	0.42	**0.74**	**0.74**	0.50	0.33	0.50	0.34	0.50	0.33
ET	0.61	0.46	0.36	0.36	0.70	0.70	0.54	0.51	0.55	0.49	0.56	0.51
GB	0.32	0.24	0.73	0.42	0.70	0.70	0.50	0.33	0.50	0.34	0.50	0.33
Average	0.46	0.38	0.63	0.47	0.70	0.69	0.55	0.46	0.57	0.47	0.56	0.47

Figure 4.4. It is clearly evident from the figure that the neural network-based classifiers dominate the feature-based classifiers in almost all the cases. It shows that the neural-based classifiers perform better than feature-based classifiers in most of the cases.



(a) Classifiers trained with `Societal-I` dataset

(b) Classifiers trained with SemEval-2013 dataset

(c) Classifiers trained with SemEval-2016 dataset

(d) Classifiers trained with Sentiment-140 dataset

Figure 4.3: Evaluation of the number of testing datasets outperforms by classifier against other classifiers built on the same datasets.

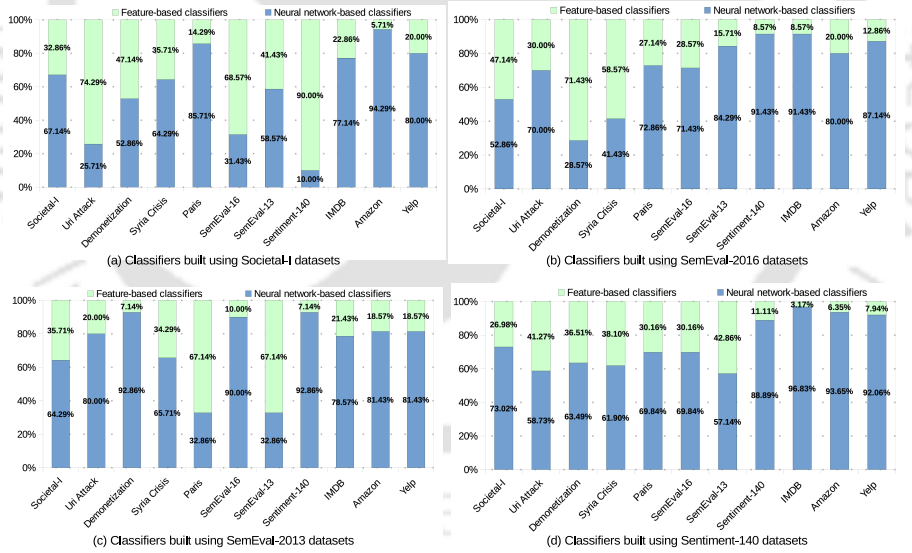


Figure 4.4: Dominance test between neural network-based and feature-based classifiers over various types of datasets

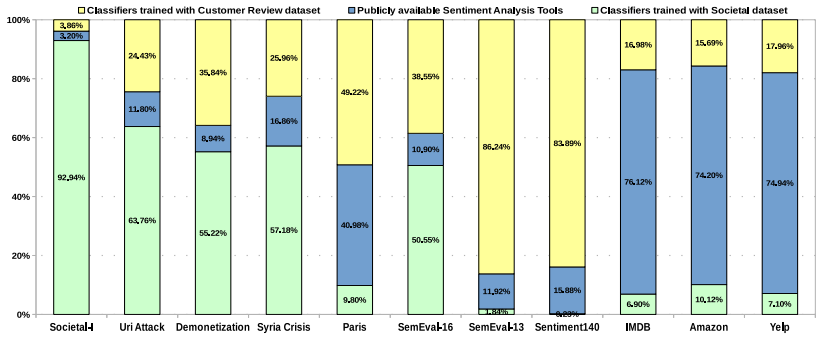


Figure 4.5: Dominance test between sentiment analysis `Tools` and `Techniques` build using societal and product review domain dataset

To answer the question *"Are the sentiment classifiers built over societal and non-societal topics compatible?"*, we further perform a dominance test of `Tools` and `Techniques` over societal topics and customer reviews in Figure 4.5. For this test, we consider the classifiers trained over Societal-I for societal topics and Sentiment-140 for customer reviews as classifiers perform relatively better over these two datasets (blue colored entries in Tables 4.5, 4.6, 4.7 and 4.8). Figure 4.5 shows the percentage of dominance of the SA `Tools` and `Techniques` for each testing datasets. It is observed that in 80% of the cases (i.e. in 8 out of 10 test datasets), locally built classifiers (either on `Societal-I` or `Sentiment-140`) dominate off-the-shelf tools with larger percentage. Interestingly, for all the testing datasets on societal domain, classifiers built on `Societal-I` dominates both the off-the-shelf tools and classifiers built on `Sentiment-140`, except for topic *Paris Agreement*. Whereas, except for `Sentiment-140` testing set, off-the-shelf tools dominate locally built classifiers (built on on both the `Societal-I` and `Sentiment-140` datasets) for customer reviews. It clearly shows that off-the-shelf tools may be suitable for sentiment analysis on customer review, but not suitable for public opinion mining on societal topics. Hence, a specialized classifier on

Table 4.9: Summary of the top performing `Tools` and `Techniques`

Classifiers	Remarks
`Tools`	
IndicoIO	This tool works best on customer reviews domain written in English language than its counterpart tools.
Rsentiment	This tool has better performance on classifying sentiment of code-mixed text than its counterpart tools.
AFINN	This tool supports code-mixed text and has better performance of sentiment classification over opinions on government policy topics than its counterpart tools.
Vader	This tool supports code-mixed text and has better performance of sentiment classification over opinions on social unrest topics than its counterpart tools.
`Techniques`	
CNN	This method have consistently performed better than its counterpart techniques.
SVM	This method have performed the best than its feature-based classifiers counterparts.

relevant societal topics may be needed for effective public opinion mining on societal topics. Based on the above observations, we summarized the best performing `Tools` and `Techniques` in Table 4.9.

4.4.3 Are public opinions on societal topics regional dependent?

To answer the question *"Can we generalize a sentiment classifier built over societal topics across different geographical locations?"*, this study investigates the performance of sentiment classifiers trained with `Societal-I` datasets for public opinion mining over events with similar nature but occurred in different regions. More specifically, this study attempts to answer the question *"Are characteristics of the public opinions on societal topics such as terror attack, political issues, etc., happened in one country different from that of another country?"* For this study,

we consider events of similar types, such as the Uri Attack and Syria Crisis, which are terrorist attack events occurring in different locations and Demonetization in India and the Paris Agreement, which are events of new government policies, to evaluate the performances of the classifiers. It is observed from Table 4.5 that the best performing classifiers are comparable (in terms of accuracy) for similar events like Uri Attack and Syria Crisis (75% with Random Forest and 79% with CNN respectively) and Demonetization and Paris Agreement (70% with Logistic Regression and 71% with CNN respectively). Further, the classifiers trained over the societal issues related to the Indian context provide encouraging performance over sentiment classification of opinions that happened in Syria and Paris. It indicates that the classifiers can capture in-variance characteristics of public sentiment over societal issues across different geographical locations. Therefore, building a generic classifier for public opinion mining over similar societal topics across different geographic areas may be feasible, sharing a common language. Though the above observations are evident from two datasets (terror attack and government policies), extended analysis on this observation with more societal topics is left as one of our future works.

4.4.4 Error Analysis

Though classifiers built over `Societal-I` outperform their counterparts over other datasets, we achieve only upto 77% accuracy. To understand the reason for low performance, we further study the characteristics of the failures in Societal-I test samples. We observe that a significant portion of the misclassified test samples are due to the following issues inherently present in tweets:

- As people are free to choose or generate hashtags without much restriction,

tweets comprising only hashtags experience out-of-vocabulary issue. If we are able to normalized out-of-vocabulary hashtags to semantically similar existing hashtags, performance of the classifiers may be enhanced.

- Irregular spelling, phonetic typing, creative writing also contribute to misclassification. Normalizing such texts can enhance classification performance. Otherwise, one needs to consider huge dataset to capture inherent pattern.

- If some keywords are dominant in one of the sentiment classes, then the classifiers are biased to that sentiment for those tweets. For example, *"@bdutt burhan vani is your head masters son?"* is a tweet annotated as neutral sentence based on content (but, it is a sarcastic comment carrying negative sentiment). However, all of the classifiers fails to capture this.

- Significant amount of the tweets on topics related to social unrest are sarcastic in nature. For example, *"@abdullah_omar @jhasanjay you must have had a grand pork party on uri n pathankot"* is an insulting comment to two individuals in the event of Uri and Pathankot terror attacks. However, this tweet carries positive sentiment if we are not aware of the entities involved. Sarcasm detection plays important role in enhancing the SA performance specially in societal topics.

- Many of the comments have stance on individuals present in tweets. A negative sentiment carrying tweet may be positive to different observers. For example *"We need next #surgicalstrike on hafeez saeed @narendramodi #baramulla #uriattack #pathankot"* may carry positive sentiment to people supporting surgical strike, but negative to people opposing surgical strike.

- The presence of large volume of code-mixed text in the `Societal` dataset, classifier not only face out-of-vocabulary issue, but also regional dependency.

From the error analysis, we observed that classifiers trained with the Societal dataset fail to classify the sentiment of tweet due to the presence of sarcasm, stance, code-mixed, and aspect-based nature of the tweet. To understand the strengths and weaknesses of the different `Tools` and `Techniques`, we further evaluate them across the following different subtasks namely `Reporting`, `Aspect-based`, `Stance`, `Sarcastic`, and `Code-mixed`.

- **Reporting:** Tweets carrying factual content are termed as `Reporting`. tweets may be classified into two categories: (i) ones which report factual or general information, such as weather report, government policies, etc. (which are generally neutral sentiment), and (ii) ones which have opinionated report, such as questioning, suggestion, stance, etc., that have sentiments. For example, we consider tweet such as *"@abpnewstv: just in: indian army has provided a 90 min video of #surgicalstrike to govt."* as factual report which have neutral or no sentiment. And tweet such as *"@firstpost: #surgicalstrikesagainstpak along #loc are a breath of fresh air writes @orsoraggiante"* are considered as opinionated report. This tweet has positive stance towards the event *Surgical Strikes*.

- **Aspect-based:** In this category, we consider those tweets that have sentiment towards any aspects present in the tweets. For example, tweet such as *"gst to spare poor make consumer king: pm"* talks about the positive aspects of GST such as *spare poor* and *consumer king*.

Table 4.10: Details of tweets annotated according to the classified categories

Categories	Positive	Negative	Neutral
Reporting	2520	516	101
Aspect	2819	547	131
Stance	378	1	7
Sarcastic	3	426	18
Codemix	312	80	23

- **Stance:** Those tweets which are biased toward a target which may or may not be present in the tweet are termed as *Stance* tweet. For example, tweets such as *"@narendramodi till now you are a leader to me and from now on you are god to me- feeling proud of #indianarmy #surgicalstrike #uriresponse"* and *"@sardesairajdeep you expose how disconnected (or prejudiced) you are from reality."* have positive stance towards *@narendramodi* and negative stance towards *"@sardesairajdeep"*.

- **Sarcastic:** Tweets with sarcastic nature are considered in `Sarcastic` category. For example, tweet such as *"@abdullah_omar @jhasanjay you must have had a grand pork party on uri n pathankot"* is an insulting comment to two individuals in the event of Uri and Pathankot terror attacks.

- **Code-mixed:** In this category, we consider those tweets that have multiple languages in a tweet.

In this study, we want to investigate the effectiveness of different off-the-shelf tools and pretrained models (classification models built over the training dataset) over tweets of different natures discussed above. We, therefore, consider one of the 10 folds (of our Societal dataset) and further annotate the sentiment of the tweets in the selected fold from the perspective of Reporting, Aspect, Stance, Sarcastic, and Code-mixed context. For example, *"@abdullah_omar @jhasanjay*

Table 4.11: Performance of the sentiment classifiers on different type of tweet categories. The boldfaces represent the classifiers outperforming other classifiers over various tweet types.

Classifiers	**Reporting**		**Aspect-based**		**Stance**		**Sarcastic**		**Code-mixed**	
	Acc	FM	Acc	FM	Acc	FM	Acc	FM	Acc	FM
MeaningCloud	0.17	0.16	0.34	0.25	0.41	0.21	0.34	0.25	0.35	0.35
SentiStrength	0.12	0.11	0.12	0.10	0.10	0.07	0.18	0.19	0.18	0.17
IndicoIO	0.41	0.28	0.57	0.34	0.69	0.28	0.15	0.13	0.38	0.27
Sentiment140	0.50	0.33	0.48	0.32	0.49	0.23	0.26	0.27	0.46	0.24
Rsentiment	0.34	0.28	0.50	0.31	0.64	0.29	0.26	0.28	0.53	0.30
AFINN	0.26	0.21	0.34	0.26	0.36	0.20	0.41	0.32	0.32	0.32
Pattern.en	0.31	0.22	0.30	0.21	0.36	0.21	0.19	0.19	0.24	0.24
Emolex	**0.52**	0.30	0.51	0.29	0.60	0.27	0.10	0.10	0.34	0.26
SentiWordnet	0.44	0.25	0.48	0.29	0.58	0.31	0.42	0.30	0.32	0.23
Vader	0.28	0.22	0.35	0.26	0.48	0.24	0.41	0.31	0.32	0.32
MLP	0.45	0.44	0.61	0.61	0.80	**0.36**	0.68	0.33	0.59	0.55
CNN	0.45	0.44	0.61	0.61	0.73	0.31	0.69	0.33	0.57	0.52
D-CNN	0.49	**0.48**	**0.67**	**0.67**	**0.86**	0.31	**0.75**	0.33	**0.63**	**0.58**
RNN	0.43	0.43	0.59	0.59	0.72	0.30	0.64	0.31	0.53	0.49
LSTM	0.45	0.45	0.61	0.61	0.72	0.29	0.66	0.30	0.53	0.49
Bi-LSTM	0.46	0.45	0.60	0.60	0.71	0.31	0.67	0.31	0.58	0.54
CNN-BiLSTM	0.42	0.40	0.58	0.58	0.68	0.31	0.86	0.33	0.56	0.40
DT	0.45	0.44	0.62	0.61	0.78	0.33	0.68	0.33	0.60	0.56
kNN	0.44	0.43	0.61	0.61	0.79	0.33	0.68	0.34	0.61	0.57
SVM (NL)	0.45	0.42	0.54	0.54	0.76	0.31	0.41	0.22	0.47	0.49
SVM	0.45	0.44	0.59	0.60	0.79	0.34	0.69	**0.34**	0.60	0.49
LR	0.46	0.45	0.61	0.61	0.80	0.34	0.69	**0.34**	0.60	0.56
RF	0.45	0.44	0.61	0.61	0.80	0.36	0.68	0.34	0.59	0.55
A-Boost (R)	0.45	0.44	0.61	0.61	0.77	0.32	0.67	0.33	0.58	0.55
A-Boost (D)	0.46	0.45	0.61	0.61	0.80	0.34	0.68	0.33	0.60	0.56
ET	0.45	0.44	0.61	0.61	0.80	0.34	0.69	**0.34**	0.60	0.56
GB	0.45	0.44	0.61	0.61	0.80	0.36	0.69	**0.34**	0.60	0.56

you must have had a grand pork party on uri n pathankot" was annotated as a positive sentiment. Considering this tweet as a sarcastic tweet, we rectify its sentiment to negative. Table 4.10 shows the statistics of annotation based on the above-discussed tweet categories. We notice that a tweet may appear in multiple categories. For example, a tweet *"@narendramodi we (youngsters) support demonetisation. but what actions you have taken against the people having black money of new currencies"* can appear in `Stance` as well as `Aspect-based` categories because the author has positive stance towards *@narendramodi* and questioning with negative sentiment towards *"black money"* as aspect.

Table 4.11 shows the performance of classifiers for different types of subcategories. It is observed from the table that classifiers trained with `Societal-I`

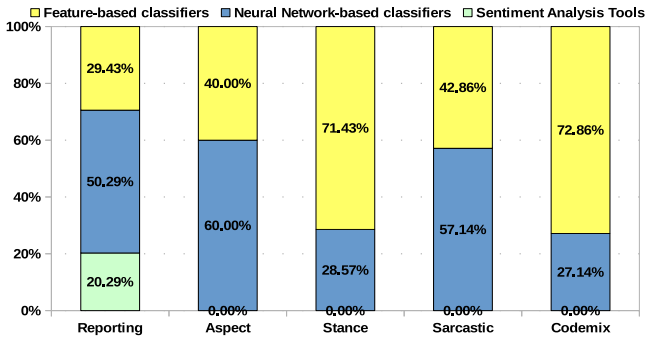


Figure 4.6: Dominance test of the sentiment classifiers over various types of tweet categories

dataset outperform the SA `Tools` in most of the cases. It is also observed that CNN based classifiers outperform other classifiers in most of the cases. Among the considered `Tools`, Emolex and Sentiwordnet perform better than other tools on classifying sentiment of `Reporting`, `Aspect-based` and `Stance` categories of tweets. AFINN and Vader perform well on `Sarcastic` and `Code-mixed` categories. Further, we perform a dominance test of the classifiers to investigate which type of classifiers (i.e. feature-based, neural network-based, or `Tools`) are suitable for SA under these subcategories. Figure 4.6 shows the dominance test of the SA classifiers. The figure shows that neural network-based classifiers dominate in three categories (Reporting, Aspect-based, and Sarcastic), while feature-based classifiers dominate on two categories (Stance and Code-mixed). It is observed that `Tools` cannot perform well on categories like Aspect, Stance, Sarcastic, and Code-Mixed. Therefore, the locally built classifiers outperform the SA `Tools` in all the categories.

4.5 Summary

In this study, we perform an empirical study to evaluate the performance of 10 publicly available sentiment analysis `Tools` and 17 state-of-the-art machines learning `Techniques` over eight datasets covering various topics of societal, customer reviews, and general discussions. From various experimental observations, it is evident that most of the off-the-shelf `Tools` are not suitable for societal topics. However, these tools have shown encouraging performance for customer reviews. Among the ten `Tools`, SentiStrength, RSentiment, AFINN, and Vader may be considered, but not to rely on, for sentiment analysis in societal topics. From the evaluation of the `Techniques`, we observe neural network-based classifiers dominate feature-based classifiers. We also note that tweets under societal issues collected from different geographical regions share common sentiment characteristics. Further, from the evaluation of the effectiveness of the `Tools` and `Techniques` over five different types of tweet categories i.e. `Reporting`, `Sarcastic`, `Aspect-based`, `Stance`, and `Code-mixed`, we observed `Techniques` have better performance than `Tools` on most of the categories. Though the classifiers trained with `Societal-I` dataset outperform the `Tools` on different types of tweet categories; still, the performance of the classifiers have low accuracy (only up to 77%). As shown in Figure 4.6, the reason for having a low performance is because of the presence of different natures of tweet such as stance, aspect-based, sarcastic, and code-mixed language tweets in the Societal dataset.

From Chapter 3 and Section 4.4.4, it is observed that people use hashtags while expressing their opinions. As people are free to choose or create hashtags with no restriction, tweets using hashtags experience out-of-vocabulary issue. Normal-

ization the out-of-vocabulary hashtags to semantically similar existing hashtags could enhance the classifier performance. Further, the sentiment representation of the tokens can improve the sentiment classification task. However, encoding sentiment information into the word embedding tampered the semantic distributions, preventing the retrieval of semantically similar sentiment polarized tokens. Therefore, the following chapter attempt to address the issues mentioned above by proposing word embedding methods that encode sentiment information while preserving token semantic information. The hashtag embedding proposed in this chapter are further used in studies in subsequent chapters.

To shine your brightest light is to be who you truly are.

Roy T. Bennett, Author of the Light in the Heart

5

SHE: Sentiment Hashtag Embedding Through Multitask learning

Recent studies have shown the importance of utilizing hashtags for sentiment analysis task on social media data. However, as the hashtag generation process is less restrictive, it throws several challenges such as hashtag normalization, topic modeling, semantic similarity, etc. Recently, researchers have tried to address the above challenges through representation learning. However, most of the studies

on hashtag embedding try to capture the semantic distribution of hashtags and often fail to capture the sentiment polarity. Further, generating a task-specific hashtag embedding can distort its semantic representation, which is undesirable for sentiment representation of hashtag. Therefore, this study proposes a semi-supervised Sentiment Hashtag Embedding (SHE) model, which is capable of preserving both semantic as well as sentiment distribution of the hashtags. In particular, SHE leverages a multitask learning approach using an Autoencoder and a Convolutional Neural Network based classifier. To assess the efficacy on hashtag embedding, we compare the performance of SHE against suitable baselines for three different tasks, namely hashtag sentiment classification, tweet sentiment classification, and retrieval of semantically similar hashtags. It is evident from various experimental results that SHE outperforms the majority of the baselines with significant margins.

5.1 Introduction

While posting opinions on social media platforms such as Twitter*, Facebook†, Youtube‡, users often use hashtags in their posts to reflect meta-information such as sentiment, emotion, topic, and entity etc. Considering its importance, many of the recent studies on opinion and social media text mining applications have given special consideration in understanding the characteristics of hashtags[133,80,64]. Understanding hashtags help in addressing various issues related to opinion and text mining tasks such as topic modeling[131,71], sentiment classification[6], sentiment lexicon generation[54,101,80], stance detection[140,81], etc. However, as people

*www.twitter.com
†www.facebook.com
‡www.youtube.com

are free to choose or generate hashtags without much restriction, it poses several challenges such as (1) normalization of hashtags representing the same entity (e.g. `#Obama`, `#BarrackObama`), (2) grouping hashtags related to similar topics (e.g. `#FacebookExit`, `#DeleteFacebook`), (3) identifying sentiment expressed by hashtags (e.g. `#AbortionIsMurder`, `#BabiesLivesMatter`) etc. To alleviate these challenges, many researchers often exploit representation learning methods like *word embedding*[48,120,31,139] and *network embedding*[96,64]. The embedding methods like Word2Vec[77], C&W[25], DeepWalk[96] represent words or hashtags into low dimensional vectors and are found to be capable of capturing semantic distribution. Such embedding methods mostly focus on learning semantic representations to be applicable on a wide range of tasks but are often found to be unsuitable for some of the domain-specific tasks like sentiment analysis[48,120,31,139]. For example, words like *good* and *bad* are semantically close* but carry different sentiment polarities. Existing studies[48,120,31,139] attempted to address the above problems using two-tier architecture; first, obtain semantic embedding using methods like Word2Vec, and second modify the semantic embedding to capture sentiment polarity using a supervised or semi-supervised sentiment classification model. Since the two steps are independent, the original semantic representation of the word may get deviated while incorporating sentiment information through separate classification model. This issue has also been observed in the studies of Tang et al.[119] and Fu et al.[31]

For tasks like sentiment lexicon generation, opinion mining, topic modeling, etc., an embedding capable of capturing both semantic distribution and sentiment polarity is desired. For example, for the hashtags like `#ModiBestPM` and

*Source: Google Word2Vec pre-trained embedding

#ModiMadeDisaster, we would be interested in recognizing that they are not only of different sentiment polarities but also related to the same person. Motivated by the above observation, this study proposes a model which is capable of preserving semantic distribution while incorporating sentiment polarity. As hashtags represent topics, sentiment, and topics having sentiment, this study choose hashtags as the target objective to perform the sentiment embedding task. However, this model applies to any type of token having semantic embedding representations. To carry out this study, we first generate pre-trained hashtag embedding using various word embedding and network embedding methods to capture semantic information of the hashtags. Thereafter, we propose a semi-supervised Sentiment Hashtag Embedding model (SHE) by exploiting multitask learning approach[105,18] to preserve semantic information of the pre-trained embedding through auto-encoder (AE) while encoding sentiment information to the pre-trained embeddings using Convolutional Neural Network (CNN) classifier simultaneously.

Over a large collection of tweets collected from Twitter, we generate sentiment hashtag embedding using the proposed model SHE. We assess the efficacy of SHE on three real-world applications, namely (i) hashtag sentiment classification, (ii) tweet sentiment classification, and (iii) retrieval of semantically similar hashtags. We compare the performance of SHE over these applications with suitable baselines. From various experimental setups for the applications mentioned above, it is evident that the proposed model SHE performs better than majority of the baselines. Further, it is also observed that SHE can be effectively used for generating sentiment hashtag lexicon for low-resource languages.

The outline of this chapter is as follow: Section 5.2 presents some of the related studies on sentiment hashtag embedding. Section 5.3 describes the detailed

framework of the proposed model SHE. We discuss the experimental setup in Section 5.4, which is followed by the experimental result and discussion in Section 5.5. Finally, Section 5.6 summarizes the study of the chapter.

5.2 Related studies

Liu et al.[64] proposes Hashtag2Vec to generate latent representations of hashtags by exploiting network embedding framework, namely DeepWalk[96]. Although their study is able to capture the semantic information of hashtags, the sentiment information is not incorporated. As our study focuses on sentiment hashtag embedding, we review some of the studies dedicated to sentiment word embedding.

Several studies have generated sentiment word embedding by exploiting semantic embedding following a two-tier approach, i.e. (i) generate semantic embedding using state-of-the-art embedding methods and (ii) encode sentiment polarity to the semantic embedding using supervised sentiment classification model[72,48,119,139,31]. For example, Mass et al.[72] have used a probabilistic topic modeling for the first time to generate semantic word embedding that is further incorporated with sentiment information by training a logistic regression over sentiment annotated documents. However, the popularity of semantic word embedding methods such as Word2Vec[77] and C&W[25] inspired the recent studies to use them as pre-trained semantic word embeddings in the first step of the above-discussed two-tier approach. To incorporate sentiment information, study in Kim[48] uses a CNN based classifier over the above-mentioned pre-trained embeddings. In a similar direction, Tang et al.[119] exploit distant supervision using emoticons for encoding the sentiment polarity. Further, studies in Ye et al.[139] and Fu et al.[31] exploit the available

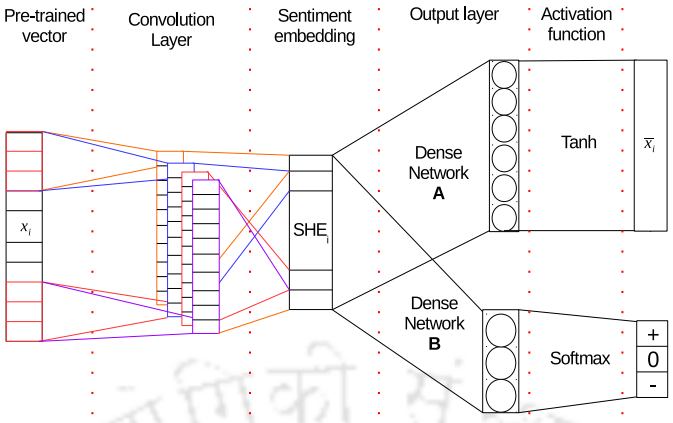


Figure 5.1: Framework of the proposed Sentiment Hashtag Embedding model using multitask learning approach

sentiment lexicons (e.g. SentiWordNet) as supervised information for generating sentiment embedding.

In all of the previous studies, semantic embedding and sentiment embedding have been seen as two independent processes. However, as discussed above, the semantic embedding can get deviated after sentiment embedding. Therefore, this study proposes to exploit a multitask learning framework[105,18] which is capable of preserving semantic characteristics while incorporating sentiment polarity by updating the model parameter jointly.

5.3 Proposed framework

5.3.1 SHE: Sentiment Hashtag Embedding

Inspired from the recent multi-task learning problems[105,66,45], where for a given input, more than one outputs are jointly learned, the proposed SHE also considers a multitask learning framework. Figure 5.1 presents a schematic diagram of the proposed model SHE through multitask learning model consisting of two learning tasks; (i) an Autoencoder (AE) for preserving semantic information, and (ii) a

classifier for incorporating sentiment polarity. To capture the latent spatial aspects of the pre-trained embedding, we use Convolutional Neural Network (CNN) in the encoding stage of the AE. In the decoding stage, a dense perceptron layer has been used. Further, the output of CNN layer (shared network) has been used as input to the softmax sentiment classifier.

Given a pre-trained semantic embedding vector $\mathbf{x}$ for a hashtag, SHE first exploits CNN to generate an intermediate representation vector $\mathbf{v}$ such that $\mathbf{v} = \textit{convolution}(\mathbf{x}, \theta)$ where θ is the convolution parameters such as the number of filters, kernel size, strides and dropout. Thereafter, $\mathbf{v}$ is passed to the decoder and the classifier units. The decoder re-generates the input vector $\mathbf{x}$ using *tanh* as activation function through dense perceptron layers and the model classifies the sentiment of $\mathbf{v}$ using *softmax* activation function.

To train the proposed model, SHE is divided into two phases. In Phase-I, the AE is trained without the softmax classifier using unlabelled hashtags in the corpus. Thereafter, in Phase-II, AE is re-trained with softmax sentiment classifier using sentiment annotated hashtags. The process of training Phase-I and Phase-II is repeated till the convergence. Once the model is trained, the sentiment embedding of a hashtag is defined by the output of the CNN layer i.e. $\mathbf{v}$.

5.3.2 Loss function for SHE

Let $\mathbf{v}$, $\mathbf{A}$, and $\mathbf{B}$ denote the output vector of shared encoding layer, weight matrix of the decoding layer, and weight matrix of the dense softmax layer respectively. Then, the output vector of the auto-encoder $\widetilde{\mathbf{x}}$ can be defined as

$$\widetilde{\mathbf{x}} = tanh(\mathbf{A}^T\mathbf{v} + \mathbf{b})$$

where $\mathbf{b}$ is the bias of the decoding layer. Similarly, output of the softmax layer $\mathbf{s}$ can be defined as

$$\mathbf{s} = softmax(\mathbf{B}^T\mathbf{v} + \mathbf{b}^{'})$$

where $\mathbf{b}^{'}$ is the bias of the softmax layer.

For a given input hashtag $\mathbf{x}_i$, the model produces two outputs i.e., $\widetilde{\mathbf{x}}_i$ from AE, $\mathbf{s}_i$ from the classifier, and generates shared $\mathbf{v}_i$ (which is the target sentiment embedding). We use mean square error for AE and cross-entropy error for the softmax classifier for learning weight matrices $\mathbf{A}$ and $\mathbf{B}$ respectively. Thus, the loss function for AE is defined as

$$\Delta_{AE} = \frac{1}{2N}\sum_{i=0}^{N}||\mathbf{x_i} - \widetilde{\mathbf{x}}_i||^2 \tag{5.1}$$

and the loss function for the softmax classifier as

$$\Delta_{CL} = -\frac{1}{N}\sum_{i=0}^{N}\sum_{c}\mathbf{t}_{ic}log(s_{ic}) \tag{5.2}$$

where c is the number of sentiment classes, $\mathbf{t}_{ic}$ is the c^{th} ground truth class index for the hashtag $\mathbf{x}_i$, N is the total number of training hashtag samples, and $\mathbf{s}_{ic}$ is the observed probability value for the c^{th} class index.

Now the loss function of the proposed model in Phase-II is defined by the sum of the two loss functions $\Delta_{AE} + \Delta_{CL}$ i.e.

$$\Delta_{SHE} = \frac{1}{2N}\sum_{i=0}^{N}||\mathbf{x_i} - \widetilde{\mathbf{x}}_i||^2 - \frac{1}{N}\sum_{i=0}^{N}\sum_{c}\mathbf{t}_{ic}log(s_{ic}) \tag{5.3}$$

This loss function is used for back-propagation through the CNN layers for esti-

mating the parameters. The rate of loss in Δ_{SHE} with respect to parameters $\mathbf{A}$ and $\mathbf{B}$ can be define as follows:

$$\frac{\partial \Delta_{SHE}}{\partial \mathbf{A_{ji}}} = 1 - tanh(\widetilde{\mathbf{x}}_i) \sum_{k=0}^{d} A_{kj} \delta_k \tag{5.4}$$

$$\frac{\partial \Delta_{SHE}}{\partial \mathbf{B_{ji}}} = s_i(1 - s_i) \sum_{c=0}^{3} B_{jc} \delta_c \tag{5.5}$$

where i and j are the neuron indices for the weight matrices $\mathbf{A}$ and $\mathbf{B}$, δ_k and δ_c are the losses in output layer for respective outputs. We then update $\mathbf{v}$ with respect to losses in $\mathbf{A}$ (say $\mathbf{v}_A$) and $\mathbf{B}$ (say $\mathbf{v}_B$). The loss in $\mathbf{v}$ is then calculated as average of $\mathbf{v}_A$ and $\mathbf{v}_B$.

$$\Delta_{\mathbf{v}} = \mathbf{v} - \mathsf{avg}(\mathbf{v}_\mathsf{A}, \mathbf{v}_\mathsf{B}) \tag{5.6}$$

5.3.3 Semi-supervised Learning

Ideally, building a sentiment hashtag classifier requires a large volume of annotated hashtags, and generating such an annotated dataset is expensive. Moreover, as people often create hashtags of their own, generating annotated datasets covering such dynamics is practically impossible. Therefore, we utilize a semi-supervised framework where a small amount of seed lexicon (publicly available lexicons) is used to influence sentiment polarity to the embedding and populate the seed lexicon.

Let $\mathcal{H}_u$ and $\mathcal{H}_l$ be the set of unlabelled hashtags and labelled hashtags respectively, where $|\mathcal{H}_u| >> |\mathcal{H}_l|$. For all $\mathbf{h}_i \in \mathcal{H}_u$, $\mathbf{t_i}$ in equation 5.2 is set to 0 (a vector with 0s elements). For the hashtags $\mathbf{h}_i \in \mathcal{H}_l$, $\mathbf{t_i}$ in equation 5.2 is set to class

probability vector i.e.,

$$\mathbf{t}_{ic} = \left\{ \begin{array}{ll} 1 & \text{if } \mathbf{h}_i \text{ belong to class c} \\ 0 & \text{otherwise} \end{array} \right\}$$

With this modification, the loss function of SHE in equation 5.3 becomes $\Delta_{SHE} = \Delta_{AE}$ for all $\mathbf{h}_i \in \mathcal{H}_u$ and $\Delta_{SHE} = \Delta_{AE} + \Delta_{CL}$ for $\mathbf{h}_i \in \mathcal{H}_l$. The set $\mathcal{H}_l$ is then expanded in semi-supervised fashion by classifying sentiment polarity of the hashtag $\mathbf{h}_i \in \mathcal{H}_u$ with a confidence higher than 95% accuracy using iterative training of SHE where input to SHE is the recent sentiment hashtag embedding.

5.4 Experimental setup

5.4.1 Dataset

This study considers a collection of approximately 973K tweets (having atleast one hashtag) crawled* from Twitter for an interval of 28^{th} April 2018 to 10^{th} September 2018. In particular, we collect tweets corresponding to (i) selected Asian countries using geo-location, (ii) trending hashtags on Twitter, and (iii) well-known user handles such as politician, news media, etc. Further, our dataset consists of three different types of tweets†, namely (i) original tweet, (ii) reply tweet, and (iii) quoted tweet. For this study, we have excluded retweets without quotes since they are same as the original tweets. Out of all the tweets considered in this study, we have 385,783 original tweets, 6,340 reply tweets, and 580,942 quoted tweets. Moreover, the crawled tweets consist of various language dynamics since we do not consider a specific target language.

*www.tweepy.org/

†https://help.twitter.com/en/using-twitter/types-of-tweets

Table 5.1: List of semantic embedding methods

Methods	Dataset type	Notation
Word2Vec (CBOW)[77]	Text	CB
Word2Vec (SkipGram)[77]	Text	SG
Deepwalk[96]	Graph	DW
Node2Vec[40]	Graph	NV
Verse[121]	Graph	VR
Multi-view embedding[76]	Graph	MVE
Hashtag2Vec[64]	Graph	HV

5.4.2 Data preparation for generating pre-trained embeddings

As discussed above, input to SHE is the pre-trained semantic embedding vectors. Thus, this study considers seven types of semantic embedding methods listed in Table 5.1 to generate pre-trained embeddings from text-based and graph-based datasets. For text-based approaches, we use the whole tweets after preprocessing such as removal of stop word, URL, etc. Further, for network-based approaches, we generate three types of undirected hashtag networks which are defined below.

- **Co-occurrence**: Two hashtags are connected if they co-occur in a tweet which could be either original tweet, quoted tweet, or reply tweet.
- **Quote-of**: Hashtag i is connected to hashtag j such that i appears in the quoted tweet, and j appears in the original tweet.
- **Reply-to**: Hashtag i is connected to hashtag j such that i appears in the reply tweet, and j appears in the original tweet.

Table 5.2 shows the characteristics of these networks. We consider Co-occurrence network for Verse[121], Deepwalk[96], Node2Vec[40], and Hashtag2Vec[64]. However, all the three hashtag networks are considered for MVE[76] to incorporate multiple views.

5.4.3 Data preparation for hashtag sentiment classification

We consider three publicly available English sentiment lexicons namely, NRC Hashtag Sentiment and Emotion Lexicons*, SentiWordnet Lexicon† and BingLiu Opinion Lexicon‡ for generating sentiment labeled hashtags. However, we do not restrict to a specific target language while training SHE. First, we consider the keywords present in these lexicons and transform them to corresponding hashtags by putting '#' as a prefix. As NRC lexicons provide sentiment score instead of sentiment labels, the hashtags are labeled in the following manner.

$$\textbf{label} = \left\{ \begin{array}{ll} \text{Positive}, & \text{if Score} \geq 0.3 \\ \text{Negative}, & \text{if Score} \leq \text{-}0.2 \\ \text{Neutral}, & \text{otherwise} \end{array} \right\}$$

The parameter of the above sentiment score is decided based on a subjective evaluation over NRC lexicons. We observe most of the *positive* sentiment words appear above 0.3 while most of the *negative* sentiment words appear below -0.2. We then choose the remaining words as *neutral*. This dataset is considered for incorporating sentiment information in proposed SHE model. There are a total of 303,194 hashtags in the considered tweet corpus, of which 17705 hashtags were matched with the sentiment lexicons mentioned above (i.e., NRC Lexicon, SentiWordnet, Opinion Lexicon, etc.). The matching hashtags are being used as seed lexicon to populate in a semi-supervised approach. Table 5.3 (column seed) shows the statistics of this collection. We refer to this collection as `#SentiLexicon` in the

*http://sentiment.nrc.ca/lexicons-for-research/
†https://sentiwordnet.isti.cnr.it/
‡https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

Table 5.2: Characteristics for different types of hashtag networks, ACC: Average clustering coefficients, CC*: Connected Component, GC*: Percentage of nodes belonging to Giant CC*

Network	#Nodes	#Edges	ACC	#CC*	GC*
Co-occurance	249,487	1,877,454	0.67	9985	90%
Quote-of	291,384	1,723,467	0.31	7673	94%
Reply-to	6,608	23,326	0.30	349	86%

Table 5.3: Data statistics of `#SentiLexicon`

Sentiment	Seed	1^{st} Iteration	2^{nd} Iteration
Positive	2937	+10	+4846
Negative	2943	+916	+17993
Neutral	11825	–	–

following sections.

5.4.4 Experimental setup for SHE

We use a 2-layer CNN encoder with 128 filters in the first layer and 64 filters in the second layer respectively. Further, we add a dropout layer of 0.2 penalty after the second CNN layer for regularization and generate the target sentiment embedding vector $\mathbf{v}$ after max-pooling. The dimension of the vector $\mathbf{v}$ is set to 64. Ideally, this dimension can be of any size. For decoding and classification phases, we consider a single dense layer using *tanh* activation function for decoder and *softmax* for classification. In this study, the training phase of the proposed model SHE converges after five iterations with 20 epochs per iteration.

5.5 Results and discussions

We investigate the performance of SHE on three tasks; (i) hashtag sentiment classification, (ii) tweet sentiment classification, and (iii) retrieval of semantically similar hashtags. To the best of our knowledge there are no works or approaches similar to the proposed SHE, therefore we consider the state-of-the-art semantic

embedding and sentiment embedding methods as our baselines. In particular, to evaluate the proposed model SHE on hashtag sentiment classification task, we compare the performance of SHE with two classes of baseline models; (i) state-of-the-art semantic embeddings, and (ii) state-of-the-art sentiment embeddings[48,139]. We consider various text and network based semantic embedding methods namely Word2Vec (CBOW (CB) and SkipGram (SG)), DeepWalk (DW), Node2Vec (NV), Verse (VR), Multi-View Embedding (MVE), Hashtag2Vec (HV) as the baseline methods. Further, sentiment embedding methods proposed in the studies[48,139] are also considered as baseline models.

5.5.1 Hashtag sentiment classification

Given a hashtag, the task is to determine its sentiment polarity. To compare the performance of SHE with its baseline classifiers, we consider a CNN classifier (with a similar setup of CNN encoder in SHE) build over `#SentiLexicon` dataset using 10-fold cross-validation. For all the hashtags in `#SentiLexicon` dataset, the corresponding embeddings are obtained from different embedding methods (baseline semantic embeddings, baseline sentiment embeddings, and SHE). Table 5.4 shows the performance of SHE over different embedding methods. We summarize the performance of SHE in the following subsections.

Are semantic embedding methods suitable for capturing sentiment information?

From Table 5.4, it is evident that among the semantic embedding methods except for Node2Vec, Hashtag2Vec, and MVE, all other embedding methods (namely CBOW, SkipGram, DeepWalk, Verse) provide classification accuracy lesser than

Table 5.4: Performance of hashtag sentiment classification using various hashtag embeddings

Approaches	Accuracy	F1 Score		
		Positive	Negative	Neutral
Baseline semantic embedding				
CB	0.37	0.11	0.12	0.53
SG	0.39	0.11	0.08	0.55
DW	0.42	0.28	0.28	0.54
NV	0.53	0.33	0.27	0.59
VR	0.49	0.36	0.34	0.61
MVE	**0.60**	0.21	0.12	0.75
HV	0.54	0.25	0.23	0.70
Baseline sentiment embedding (SE)				
CB+SE	0.43 (+16.21%)	0.21 (+90.91%)	0.26 (+116.67%)	0.57 (+7.55%)
SG+SE	0.35 (-10.26%)	0.13 (+18.18%)	0.14 (+75%)	0.51 (-7.27%)
DW+SE	0.40 (-4.76%)	0.00 (-100%)	0.00 (-100%)	0.57 (+5.55%)
NV+SE	0.54 (+1.89%)	0.32 (-3.03%)	0.30 (+11.11%)	0.68 (+15.25%)
VR+SE	0.48 (-2.04%)	0.32 (-11.11%)	0.33 (-2.94%)	0.60 (-1.64%)
MVE+SE	**0.64** (+6.67%)	0.03 (-85.71%)	0.11 (-8.33%)	0.79 (-5.33%)
HV+SE	0.62 (+14.81%)	0.37 (+48%)	0.26 (+13.04%)	0.75 (+7.14%)
SHE without Lexicon expansion				
CB+SHE	**0.75** (+103%)	0.62 (+464%)	0.63 (+425%)	0.84 (+58%)
SG+SHE	0.65 (+67%)	0.49 (+345%)	0.39 (+388%)	0.78 (+42%)
DW+SHE	0.59 (+40%)	0.19 (-32%)	0.14 (-50%)	0.74 (+37%)
NV+SHE	0.52 (-2%)	0.32 (-3%)	0.32 (+19%)	0.65 (+10%)
VR+SHE	0.52 (+6%)	0.35 (-3%)	0.36 (+6%)	0.64 (+5%)
MVE+SHE	**0.76** (+27%)	0.69 (+229%)	0.59 (+392%)	0.84 (+12%)
HV+SHE	0.61 (+12.96%)	0.26 (+4%)	0.22 (-4.35%)	0.76 (+8.57%)
SHE with Lexicon Expansion				
CB+SHE	**0.76** (+1%)	0.69 (+11%)	0.59 (-6%)	0.84
SG+SHE	**0.79** (+22%)	0.70 (+43%)	0.64 (+64%)	0.86 (+10%)
DW+SHE	0.60 (+2%)	0.16 (-16%)	0.13 (-7%)	0.75 (+1%)
NV+SHE	0.54 (+4%)	0.32	0.30 (-6%)	0.68 (+5%)
VR+SHE	**0.78** (+50%)	0.77 (+120%)	0.75 (+108%)	0.80 (+25%)
MVE+SHE	**0.79** (+4%)	0.76 (+10%)	0.79 (+34%)	0.81 (-4%)
HV+SHE	0.63 (+3.28%)	0.36 (+38.46%)	0.22	0.76

0.5. Further, network embedding methods perform better than its text-based counterparts. It also shows that co-occurring characteristics of hashtags can capture sentiment information better than co-occurring characteristics of the running text. The best performance is obtained using MVE with an accuracy of 0.6. Further, among all the network-based embedding methods, MVE performs the best. This infers that in addition to co-occurrence relation, other contextual relations such as Quote-of and Reply-to help in capturing better sentiment information.

Effectiveness of incorporating sentiment information

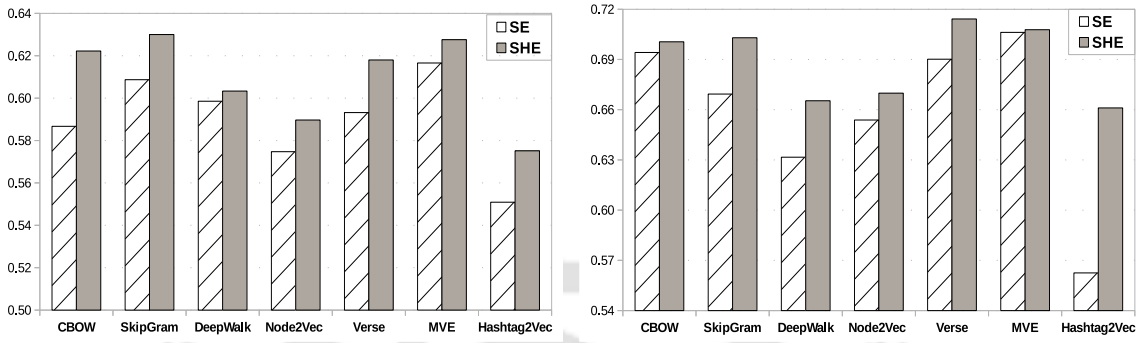
In this section, we investigate the performance of incorporating sentiment information using SHE and its baseline counterparts[48,139]. We incorporate sentiment information over all the semantic embedding listed in Table 5.1. From Table 5.4, it is observed that sentiment embedding provides better classification accuracy and F1 measures for both SHE and baseline sentiment embedding (SE) in majority of the cases over baseline semantic embeddings. It is evident from Table 5.4 that the proposed model SHE improves the classification performance of all the embedding methods except Node2Vec. We achieve the best accuracy upto 0.76 for MVE+SHE which is approximately 27% and 19% improvement over semantic embedding using MVE and sentiment embedding using MVE+SE respectively. Further, the best baseline sentiment embedding i.e. MVE+SE provides an improvement of approximately 7% over the semantic embedding using MVE. Thus, it can be inferred that incorporating sentiment information improves the quality of sentiment hashtag embedding.

Effectiveness of sentiment lexicon expansion

This section investigates the efficacy of sentiment lexicon expansion with SHE for hashtag sentiment classification. We expand the `#SentiLexicon` using the framework discussed in Section 5.3.3. Table 5.3 presents the number of expanded lexicons for each iteration of the expansion process. As shown in Table 5.4, classification performance is further enhanced with the expanded lexicon for all the frameworks. It is observed that with lexicon expansion, SHE improves the classification accuracy by 4% (for MVE+SHE) over SHE without lexicon expansion.

Table 5.5: Characteristics of the Experimental Datasets

Dataset	Positive	Negative	Neutral	Total
SemEval-2013	5115	2017	6099	13231
SemEval-2016	1296	2491	276	4063



(a) SemEval 2013 Dataset (b) SemEval 2016 Dataset

Figure 5.2: Performance of tweet sentiment classification using SE and SHE

This observation indicates that semi-supervised sentiment lexicon expansion helps in generating a better quality of sentiment hashtag embedding.

5.5.2 Effect of SHE in tweet sentiment classification

This section investigates the effect of SHE in determining the sentiment polarity of a tweet. For this task, we use two Twitter datasets namely SemEval-2013* and SemEval-2016†. Table 5.5 shows the statistics of these datasets. The tweet sentiment classification framework is inspired from the study in[48]. Since this study focuses on learning sentiment representation of hashtags present in the tweets (not on building a tweet sentiment classification model), we compare the performance of SHE with corresponding baseline sentiment embeddings (for example CBOW+SE vs CBOW+SHE). For utilizing sentiment hashtag embeddings such as SE and SHE, we treat each keyword present in tweets as hashtags. Further, we

*https://www.cs.york.ac.uk/semeval-2013/task2/
†http://saifmohammad.com/WebPages/StanceDataset.htm

MVE	MVE+SE	MVE+SHE
facebook: twitter, june30, speech, intersections, google, ig, linkedin, jobsearch, eaircraft, samsungelectronics **delefacebook:** bigpharma, kkk, rhinos, wematter, angrydemocrats, obstructionists, nzpol, whitegenocide, jnj, wearethepeople **presidenttrump:** russians, strzokhearing, russianhacking, mueller, babytrump, nunestapes, helsinkicalling, strzok, trumpuk, helsinkisummit2018	**facebook:** linkedin, google, crime, off, technical, fail, backup, traffic, die, frustrated **deletefacebook:** bigpharma, rhinos, wematter, angrydemocrats, obstructionists, nzpol, whitegenocide, jnj, wearethepeople, rapetorturekill **presidenttrump:** russians, strzokhearing, russianhacking, mueller, babytrump, nunestapes, helsinkicalling, strzok, trumpuk, helsinkisummit2018	**facebook:** need, linkedin, google, technical, crime, animal, onlineshopping, hacks, scams, injury **deletefacebook:** putininstalledpuppet, pardonedforwarcrimes, fda, arresttreasonouspence, singlepayer, trumpisanazi, liers, repealandreplace, firethegop, oneofyou **presidenttrump:** russians, va07deservesbetter, redpill, trumptrain, lisapage, bannon, trumpbabyblimp, nunestapes, boycottnfl, northkoreansummit

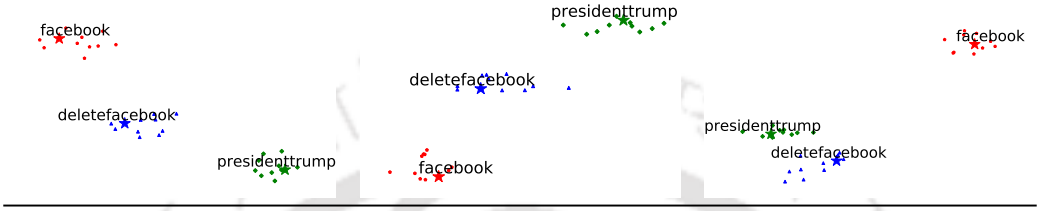


Figure 5.3: Distribution of hashtags retrieved for the queries defining *Delete Facebook Campaign*; the symbol star denotes the query and the dots denote the retrieved hashtags for each query.

do not consider new hashtags of a tweet which are not present in the vocabulary of hashtags considered by the experimental dataset for SHE. We compare the performance of SHE with the baseline sentiment embeddings on sentiment classification of tweets. It is observed that in majority of the cases, tweet sentiment classification with SHE outperforms its counterpart sentiment embedding. Over the SemEval-2013 dataset, MVE+SHE outperforms others with an accuracy of 0.63 whereas, in SemEval-2016 dataset, Verse+SHE outperforms others with an accuracy of 0.71.

5.5.3 Effectiveness of retrieving semantically similar hashtags

In this section, we assess the capability of SHE in retrieving semantically similar hashtags for the queries related to event representations. To investigate the retrieval performance, two cases are reported in Figures 5.3 and 5.4. In Figure 5.3,

MVE	MVE+SE	MVE+SHE
thailandrescue: navyseals, thamluangcaverescue, caverescue, cavethailand, samangunan, thaicaverescuemission, thairescue, samankunan, cavesrescuethailand, 13strong **keralaflood:** keralafloodrelief, keralafloods2018, keralafloodshelpline, keralarelieffund, keralarains, fishermen, keralafloodrescue, kerelafloodrelief, keralafloods, nammudekeralam **rescuer:** impossiblejob, stopkillingyemen, ittakesavillageliterally, thilandcaverescue, luangcave, nationnews, aspenbrave, basstageonstage, heroeshelpingheroes, sumangunan	**thailandrescue:** navyseals, thamluangcaverescue, caverescue, cavethailand, samangunan, thaicaverescuemission, thairescue, samankunan, cavesrescuethailand, 13strong **keralaflood:** keralafloodrelief, keralafloods2018, keralafloodshelpline, keralarelieffund, keralarains, fishermen, keralafloodrescue, kerelafloodrelief, keralafloods, nammudekeralam **rescuer:** impossiblejob, stopkillingyemen, ittakesavillageliterally, thilandcaverescue, luangcave, nationnews, aspenbrave, basstageonstage, heroeshelpingheroes, sumangunan	**thailandrescue:** 13strong, samangunan, thamluangcaverescue, thaicavesrescuemission, thaiboys, caverescue, navyseals, tailandiarescate, caverescuethailand, samankunan **keralaflood:** keralarelieffund, keraladonationchallenge, indianarm, wahegguru, donateforkerala, cmrelieffund, leftlobby, keralarains, sikhforhumanity, valueyourmoney **rescuer:** 121stevac, maltouch, allboysandcoachout, heroeshelpingheroes, ninenetwork, thamluangcavesrescue, semangat, ch3thailandnews, subwaylive, thaicaverescue11th

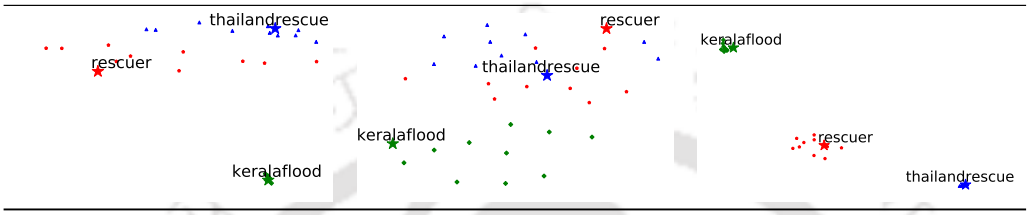


Figure 5.4: Distribution of hashtags retrieved for the queries defining the events *Tham Luang Cave Rescue* and *Kerala Flood 2018*; the symbol star denotes the query and the dots denote the retrieved hashtags for each query.

distribution of retrieved hashtags for the event *Delete Facebook Campaign*[*] has been presented. In these plots, we attempt to capture similarity between the *Delete Facebook Campaign* and *President Donald J Trump* by submitting hashtags `#DeleteFacebook`, `#Facebook`, and `#PresidentTrump`. Cosine similarity between two embedding vectors has been used as the retrieval model. Since MVE consistently outperforms other embedding methods on the hashtag sentiment classification task (refer Table 5.4), we consider MVE, MVE+SE, and MVE+SHE for comparing retrieval performance of the proposed SHE model. It is observed from the plots that the retrieved hashtags using baseline embeddings (i.e. MVE and MVE+SE) are not able to capture the semantic similarity between `#DeleteFacebook` and `#PresidentTrump`. However, the proposed model (i.e. MVE+SHE) is able to capture better semantic similarity and plots them closer. It

[*] `#DeleteFacebook` and `#PresidentTrump` were used by several users to express their sentiments on the exploitation of Facebook data by the current President of USA, Mr. Donald J Trump, for his presidential election campaign. https://www.vox.com/policy-and-politics/2018/3/23/17151916/facebook-cambridge-analytica-trump-diagram

can also be seen from the figure that the hashtags like #PutinInstalledPuppet, #TrumpisaNazi, #Russians, etc. are retrieved among the top results for the query #DeleteFacebook.

Further in Figure 5.4, we investigate the distribution of retrieved hashtags for two semantically similar events, namely *Tham Luang Cave Rescue*[*] and *Kerala Flood 2018*[†] by submitting following three queries #ThailandRescue, #KerelaFlood, and #Rescuer. It is observed that all the three embedding methods (i.e. MVE, MVE+SE, and MVE+SHE) retrieved convincing semantically similar hashtags for individual queries (refer top ten retrieve hashtags in Figure 5.4). However, the plots obtained using MVE and MVE+SE are more scattered as compared to MVE+SHE. It indicates that MVE+SHE provides better cluster-ability (i.e. better event representation) as compared to MVE and MVE+SE.

5.5.4 Sentiment hashtag lexicon in non-English languages

The proposed embedding method is found to be capable of capturing hashtags with similar sentiment and semantic representation across different languages. To validate these observations, we retrieve non-English hashtags using the queries like #ThailandRescue, #RahulHugPM, and #ModiSarkar and classify the sentiments using SHE embedding as discussed in Section 5.5.1. Some of the results are shown in Figure 5.5. These results consist of hashtags written in Hindi, Tamil, Thai, Japanese, etc. It also has phonetically typed hashtags in Hindi language. The sentiment of hashtags written in native script text are evaluated using Google

[*]Thirteen boys stuck in Tham Luang Cave https://en.wikipedia.org/wiki/Tham_Luang_cave_rescue

[†]The worst flood in Kerala after nearly a century https://en.wikipedia.org/wiki/2018_Kerala_floods

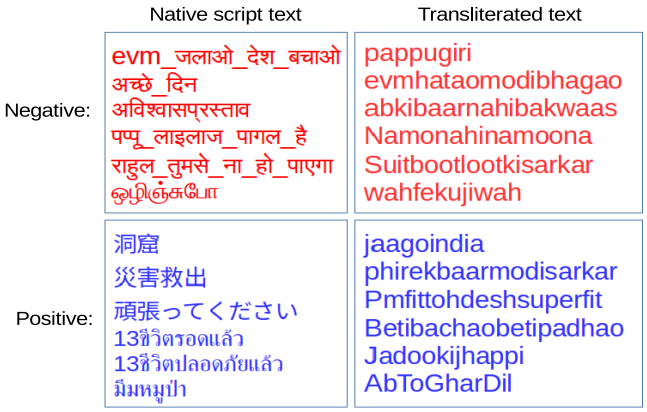


Figure 5.5: Sentiment polarity of few of the popular hashtags in non-English languages identified using SHE. Red color indicates negative sentiment; Blue color indicates positive sentiment

Translate*. Further, the transliterated hashtags written in Hindi language are manually evaluated through Hindi speakers. From the experimental results presented in Figure 5.5, it is evident that SHE can capture sentiment information for hashtags in non-English languages as well. Therefore, SHE can be used as a method to generate sentiment hashtag lexicon for non-English languages using seed sentiment lexicons discussed in Section 5.4.3.

5.6 Summary

This study proposes a novel semi-supervised Sentiment Hashtag Embedding (SHE) model capable of encoding sentiment polarity while preserving the semantic characteristics of hashtags. In particular, we exploit multitask learning approach through Autoencoder and Convolutional Neural Network classifier to train the proposed SHE model. From various experimental evaluations, it is observed that SHE yields robust hashtag embeddings and performs better than state-of-the-art baselines. It is also observed that SHE can be effectively used for various tasks like

*https://translate.google.com/

hashtag sentiment classification, tweet sentiment classification, hashtag retrieval, and sentiment hashtag lexicon generation for non-English languages.

This study observed that sentiment hashtag embedding trained using network representation captured better sentiment information than text sequences. Further, the multi-view representation of hashtags has captured better semantics of hashtags than the homogeneous hashtag network. In the next chapter, to enhance the performance of the sentiment classification task, each tweet is represented as a heterogeneous multi-layer network considering the tweet components, i.e., keyword, mention, and hashtags as the respective layers of the heterogeneous network.

Don't build links. Build relationships.

Rand Fishkin, CEO & Co-Founder of Moz

6

Sentiment Analysis of Tweets using Heterogeneous Multi-layer Network Representation and Embedding

Sentiment classification on tweets often needs to deal with the problems of under-specificity, noise, and multilingual content. This study proposes a heterogeneous multi-layer network-based representation of tweets to generate multiple represen-

tations of a tweet and address the above issues. The generated representations are further ensembled and classified using a neural-based early fusion approach. Further, we propose a centrality aware random-walk for node embedding and tweet representations suitable for the multi-layer network. From various experimental analysis, it is evident that the proposed method can address the problem of under-specificity, noisy text, and multilingual content present in a tweet and provides better classification performance than the text-based counterparts. Further, the proposed centrality aware based random walk provides better representations than unbiased and other biased counterparts.

6.1 Introduction

With the growing popularity of Twitter, sentiment analysis of tweets has drawn the attention of several researchers from both academia and industry in recent times. Unlike other regular texts, sentiment analysis on Twitter text poses plenty of challenges because of various characteristics such as (i) under-specificity due to text limits, (ii) free-form writing such as the presence of user-defined hashtags, mentions, emoticons, (iii) noisy texts due to the presence of short-form, long-form, multilingual, transliterated text, misspelling. Researchers try to address these problems by adopting various methods like task-specific representation learning[113,97,30,119,48], incorporating additional information such as hashtags[6,101], relationship between users[148], multi-source information[149], ensembling[5,8,130], etc.

This study proposes a novel approach to handle the above issues using a heterogeneous multi-layer network representation of a tweet. A multi-layer network is a network formulated by connecting different layers of networks. For example,

a heterogeneous multi-layer network can be formed by connecting layers of networks of mentions, hashtags, and keywords. Multi-layer networks have shown to provide promising performance in other tasks like community detection and clustering[42,70], node classification[60,151,34], representation learning in graphs[20,142,89]. A tweet or a collection of tweets can be represented by a multi-layer network. An advantage of using network-based representation is that a network can be expanded by adding nodes or shrunk by removing nodes. The motivations of using a multi-layer network in this study are as follows. (i) The semantic relation between keywords, hashtags, and mentions can be captured by applying an effective network embedding method. (ii) The noise and under-specificity can be reduced by expanding the network with related nodes or by shrinking the network after removing the unrelated nodes. Further, the co-occurring keywords, hashtags, and mentions often share semantic relationships[132,133,100,131].

This study has four major contributions. First, it transforms a tweet into a multi-layer network. Second, it proposes a centrality* aware random walk over the multi-layer network. Third, it generates multiple representations of a tweet using the proposed centrality aware random walk and builds an early-fusion based neural sentiment classifier. Fourth, it also addresses under-specificity and noisy text for sentiment classification by expanding or shrinking the network representing the tweets. As such, sentiment classification is a domain-dependent task[46]. Therefore, we evaluate the proposed method over datasets in different domains. From extensive experimental evaluations, the proposed method is found to outperform its counterparts in the majority of the cases. To the best of our knowledge, this study is the first of its kind to investigate sentiment classification task by

*Prominence of a node in a network

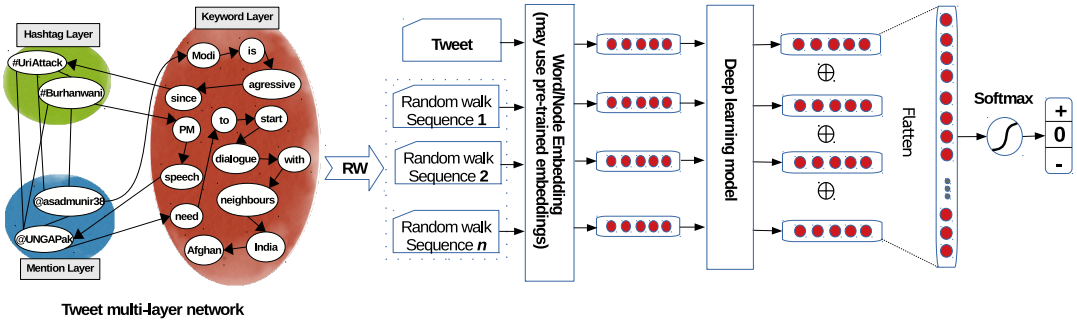
transforming tweet into a heterogeneous multi-layer network.

The rest part of the chapter is organized as follows. Section 6.2 presents the literature related to this study. Section 6.3 presents the proposed framework. The experimental setup is described in Section 6.4. The results and observations are analyzed in Section 6.5. Finally, Section 6.6 summarizes the study of this chapter.

6.2 Related studies

Sentiment analysis is an old research area. Initial work on sentiment classification can be traced back as early as 2000[123,95,124]. There have been several paradigm shifts in sentiment analysis methods from statistical methods[123,95,124] to rule-based[99], to lexicon-based[118,9,78], to feature-based[53,10], to deep neural network[48,109]. Majority of the recent studies focus on the application of neural network models. Therefore, this section briefly reviews a few of the recent and related studies which have exploited graph and neural models.

Violos et al.[129] use a homogeneous network known as *word graph* to represent a document by connecting co-occurring words in the document. Three different networks are created for positive, negative, and neutral classes using the documents in respective classes. Using these networks, a document is represented by a three-dimensional vector defined by the three sentiment classes. The elements of the vector correspond to the similarity of the word graph of the document and the word graph of the respective sentiment class. The vector thus obtained is used for classifying the document. Similarly, Bijari et al.[11] construct co-occurrence word-graph of a document collection and generate word embedding using Node2Vec[40]. The embeddings thus obtained are used to represent words in the text and build



Tweet: *@asadmunir38 Modi is agressive since #UriAttack, #BurhanWani & PM speech @UNGAPak needs to start dialogue with neighbours India, Afghan*

Figure 6.1: Proposed heterogeneous multi-layer network based tweet sentiment classification framework

a classifier using the Convolution Neural Network (CNN) model. Further, in the studies [41,148], the advantages of exploiting the relationship between keywords, sentiment, products and users have also been evident in sentiment analysis.

In recent times, deep learning based models are extensively used for sentiment classification. To mention few of them, authors in [44,27,109,29,48] use CNN, authors in [136,65] have used Long Short Term Memory (LSTM), Bidirectional LSTM (Bi-LSTM), while authors in [88,23] uses a combination of convolution and recurrent based neural network models. Further, studies in [5,8] uses neural ensemble models to combine different representation of text.

6.3 Proposed framework

As mentioned earlier, the proposed method has four distinct components; (i) representation of a tweet or collection of a tweet using a multi-layer network, (ii) centrality aware random walk over the multi-layer network, (iii) tweet classification using multiple representations generated from the multi-layer network of a tweet, and (iv) reduction of noise in a tweet by expanding or shrinking network. This section discusses the details of these components. Figure 6.1 shows a high-

level schematic diagram of the proposed model using a heterogeneous multi-layer network.

6.3.1 Representation of tweets using multi-layer network

A L-layer network $\mathbf{G}$ is defined by $(\mathbf{V}, \mathbf{E}, \mathcal{L})$ where $\mathcal{L}$ denotes the set of layer indices $\{1, 2, ..., L\}$, $\mathbf{V} = \{\mathbf{V}^1 \cup \mathbf{V}^2 \cup ... \cup \mathbf{V}^L\}$, $\mathbf{V}^i$ denotes the set of vertices in layer i of the network, $\mathbf{E}$ denotes the set of edges. Considering three important components of a tweet, the proposed multi-layer network is formed with three layers i.e., *hashtag, mention* and *keyword* as $\{H, M, K\}$. To capture both the co-occurrence and sequential characteristics of keywords, hashtags and mentions in a tweet, the proposed network consists of both directed and undirected edges. An edge $e_{x,y} \in \mathbf{E}$ is directed if x and y occur sequentially next to other in a tweet where, i) $x, y \in V^K$ or ii) $x \in V^K$ and $y \in \{V^H \cup V^M\}$ or iii) $x \in \{V^H \cup V^M\}$ and $y \in V^K$. Whereas, an edge $e_{x,y} \in \mathbf{E}$ is undirected if $x, y \in \{V^H \cup V^M\}$ co-occur in a tweet. An example of the proposed multi layer network for the tweet *"@asadmunir38 Modi is agressive since #UriAttack, #BurhanWani & PM speech @UNGAPak needs to start dialogue with neighbours India, Afghan"* is shown in Figure 6.1. Edge set $\mathbf{E} = \{\mathbf{A} \cup \mathbf{B}\}$ which comprises of a set of intra-layer adjacency matrices $\mathbf{A} = \{\mathbf{A}^1, \mathbf{A}^2, ..., \mathbf{A}^L\}$ with matrix $\mathbf{A}^i \in \mathcal{R}^{N^i \times N^i}$ in each layer i. A set of bipartite matrices $\mathbf{B}^{i,j} \in \mathcal{R}^{N^i \times N^j}$ represents cross-layer association between layer i and layer j. For our tweet multi-layer network, we have three layers $\mathbf{A} = \{\mathbf{A}^H, \mathbf{A}^M, \mathbf{A}^K\}$ and five types of bipartite associations $\mathbf{B} = \{\mathbf{B}^{HM}, \mathbf{B}^{MK}, \mathbf{B}^{HK}, \mathbf{B}^{KM}, \mathbf{B}^{KH}\}$. This kind of complex networks can also be viewed as one flattened representation in form of

supra-adjacency matrix S, with total nodes $N = |\mathbf{V}^H| + |\mathbf{V}^M| + |\mathbf{V}^K|$,

$$\mathbf{S}_{N\times N} = \begin{bmatrix} \mathbf{A}^H & \mathbf{B}^{HM} & \mathbf{B}^{HK} \\ \mathbf{B}^{MH} & \mathbf{A}^M & \mathbf{B}^{MK} \\ \mathbf{B}^{KH} & \mathbf{B}^{KM} & \mathbf{A}^K \end{bmatrix} \tag{6.1}$$

The intra-layer associations $\mathbf{A}$s are on the main-diagonal, and the cross-layer connections $\mathbf{B}$ are on the off-diagonal elements of $\mathbf{S}$. Further, $\mathbf{A}^K, \mathbf{B}^{HK}, \mathbf{B}^{KH}, \mathbf{B}^{MK}, \mathbf{B}^{KM}$ are asymmetric matrices and other matrices of $\mathbf{S}$ are symmetric. A tweet or a collection of tweets can be represented as a multi-layer network, as discussed above. A global multi-layer network is represented by combining all six relations of nodes from the whole tweet corpus to capture the insight properties of the nodes via node embedding (refer Section 6.4.2).

6.3.2 Centrality aware random-walk with restart for heterogeneous multi-layer network

To generate random walk sequences from the proposed multi-layer tweet network, we extend the random walk followed in PageRank[14] algorithm. Given a row stochastic adjacency matrix $\mathbf{A}$ of a network, the PageRank of the nodes in the network can be defined as the following vector.

$$\vec{\pi}_{t+1} = (1-\delta)\mathbf{A}\vec{\pi}_t + \delta\vec{\pi}_0 \tag{6.2}$$

where $\vec{\pi}_t$ is the stationary probability distribution vector that depicts the probability with which a random walker would stay in a particular node at time t. The restart probability $\delta \in [0,1]$ denotes the probability of jumping to a random node

and $\vec{\pi}_0$ is the initial stationary probability vector.

As in Li et al.[61], the above random-walk can be extended to our tweet multi-layer heterogeneous network in the following manner. If $\lambda \in (0,1)$ is the probability that a random-walker jumps to a different layer while surfing, in presence of L number of layers and considering jumping to any of the remaining layers is equiprobable, the transition probability $\mathbf{M}$ aka column-normalized supra-adjacency matrix $\mathbf{S}$ in Equation 6.1, is modified as,

$$\mathbf{M} = \begin{bmatrix} (1-\lambda)A^H & \frac{\lambda}{L-1}B^{HM} & \frac{\lambda}{L-1}B^{HK} \\ \frac{\lambda}{L-1}B^{MH} & (1-\lambda)A^M & \frac{\lambda}{L-1}B^{MK} \\ \frac{\lambda}{L-1}B^{KH} & \frac{\lambda}{L-1}B^{KM} & (1-\lambda)A^K \end{bmatrix} \tag{6.3}$$

That is, for a node, if its bipartite association exists, a random-surfer can stay in the same layer with probability $(1-\lambda)$ or transit to a different layer with probability $(\frac{\lambda}{L-1})$. Now, Equation 6.2 can be re-written as follows,

$$\vec{\pi}_{t+1} = (1-\delta)\mathbf{M}\vec{\pi}_t + \delta\vec{\pi}_{rs} \tag{6.4}$$

where $\vec{\pi}_{rs} = \begin{bmatrix} \eta_H.\vec{\pi}_0^H \\ \eta_M.\vec{\pi}_0^M \\ \eta_K.\vec{\pi}_0^K \end{bmatrix}$, η_i denotes the importance of layer i, $\vec{\pi}_0^i$ denotes the initial stationary distribution of nodes in layer i and $\sum_{i\in\{H,M,K\}}\eta_i = 1$. And, $\vec{\pi}_t \in \mathcal{R}^{(N^H+N^M+N^K)}$ is the stationary probability distribution of a random surfer on the heterogeneous multi-layer network at time t.

In this study, we propose to personalize the above PageRank algorithm using the global importance of nodes in the proposed heterogeneous multi-layer net-

work. In Equation 6.4, $\vec{\pi}_{rs}$ the restart probability vector is interpreted as layer importance weighted over the centrality based initial stationary probabilities of nodes. This interpretation needs not only the node centrality scores but also the layer importances. MultiRank[103], a centrality estimate for multiplex networks* formulated using a modified version of PageRank algorithm, can estimate both the node centrality scores as well as the layer influences. MultiRank uses a layer-influence weighted aggregated adjacency matrix and a weighted bipartite matrix that relates nodes with layers to determine the node and layer centrality scores simultaneously. We specifically change the definition of these two matrices to customize the MultiRank algorithm for estimating the centrality scores over the heterogeneous multi-layer network representation of tweets. As we calculate the centrality scores, we modify $\vec{\pi}_{rs}$ of Equation 6.4 by replacing each η_i with respective influence score of layer i and each initial stationary vector $\vec{\pi}_0^i$ with node centrality scores in layer i.

In the customized MultiRank algorithm, we have tuned free-parameters (as described in the original paper) while calculating the centrality scores – i) to suppress or enhance the contribution of low-centrality nodes, ii) to take into account the elite layers that contain a few highly central nodes, iii) to or not to normalize layer influences by weighted layer in-strength. We have tuned the restart parameter in MultiRank and multi-layer random walks in the range $\in [0.5, 0.85]$. In this study, the MultiRank algorithm and multi-layer random walks gave the best performance by setting the restart parameter to 0.5 and 0.85, respectively. Furthermore, the average number of tokens per tweet present in our training dataset

*Multiplex network[52] is a special case of a multi-layer network that has the same set of nodes exhibiting distinct relations in different layers.

Table 6.1: Different embedding and neural methods

Node embedding methods
FastText Embedding (FT)[13]
Multi-View Embedding (MVE)[102]
Multiplex Network Embedding (MNE)[142]
Sentiment Hashtag Embedding (SHE)[113]

* The embedding dimension is of 128 size. Same hyper-parameter as suggested in the literature.

Deep-learning models	Hyper-parameter
Convolution Neural Network (CNN)	3 Kernels, 128 #Filters, *ReLu* Activation Function
Bidirectional Long Short Term Memory (Bi-LSTM)	64 LSTM Units, *ReLu* Activation Function

is 29, so we have hypothetically set the walk-length at 30. We set the number of walks at 10. All the free parameters are tuned based on end-task performance.

6.3.3 Classification of tweets represented with a multi-layer network

Let G_i be the multi-layer network representing a tweet T_i. Over this network, we generate n number of node sequences $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, ..., \mathcal{S}_n\}$ by using the above proposed random walk. Each node sequence is maintained to have a length of m nodes. With n number of random sequences and the original tweet, we have $(n+1)$ sentences to represent the tweet T_i. Each word in these sentences can be represented using a vector obtained from an appropriate embedding method. This study has considered different embedding methods, as listed in Table 6.1, trained over a large collection of tweets.

For each node sequence $\mathcal{S}_i$, we apply a neural model (Bi-LSTM[23] and CNN[48]) to generate a representation of the sequence $\mathcal{S}_i$. The last hidden state output obtained after passing the node sequences to Bi-LSTM represents the sequence $\mathcal{S}_i$. While, the vector obtained after applying the pooling step in CNN represents the sequence $\mathcal{S}_i$. Thus, we obtained $(n+1)$ vectors for each tweet. We concatenate these $(n+1)$ vectors and feed it to a feed-forward dense layer with three neurons

(each for positive, negative, and neutral) and classify the sentiment of the tweet using softmax activation function in the output layer as shown in Figure 6.1. We use Keras* deep learning framework for building our proposed model.

We calculate the error loss (Δ) for the classifier using the well-known cross-entropy loss as,

$$\Delta = -\frac{1}{l}\sum_{i=1}^{l}\sum_{c}\mathbf{t}_{ic}log(s_{ic}) \tag{6.5}$$

where c is the number of sentiment classes, $\mathbf{t}_{ic}$ is the c^{th} ground truth class for the tweet, l is the total number of training samples, and $\mathbf{s}_{ic}$ is the predicted probability on sample i for the c^{th} class.

6.3.4 Network expansion and shrinking

One of the motivations of using the multi-layer network for representing a tweet lies in its flexibility to expand or shrink the network. Given a set of existing nodes in a tweet-network as query nodes, the idea is to identify the most related nodes or most noisy nodes by exploiting a multi-layer network of a global tweet collection. We consider the most central and most similar neighboring nodes of the query nodes as potential expansion candidates. To reduce the search space, we first select the top k query nodes ranked by the nodes' centrality scores in the tweet network view. The centrality scores of the nodes are calculated from the whole tweets collection. We then find neighbors of the selected nodes and rank them using a weighted combination of similarity and centrality score using the

*https://keras.io

scoring function defined below:

$$Score(v) = \sum_{u \in N_v} \alpha \cdot sim(v,u) + (1-\alpha) \cdot centrality(u) \tag{6.6}$$

where N_v denotes neighbouring nodes of v, $sim(v,u)$ denotes cosine similarity using node embeddings of v and u, and $centrality(u)$ denotes centrality score of node u in global network. In this study, we take equal weights of cosine and centrality score by setting $\alpha = 0.5$. Top neighbouring nodes are selected using the above scoring function and added to the network in their respective layers using the edge policy discussed in Section 6.3.1.

The above node expansion method finds new nodes having semantic relation with the query nodes. However, for the sentiment analysis task, we are interested in adding only sentiment bearing nodes by selecting only those nodes having the dominant sentiment class among the selected nodes for expansion. While, the rest of the nodes with less dominating sentiment classes are removed from the tweet network. The Sentiment Hashtag Embedding (SHE) method proposed in[113] is used to estimate the sentiment orientation of a node. We have used the same experimental setup as described in the literature.

6.4 Experimental Setup

6.4.1 Dataset

This study considers the locally annotated dataset named as `Societal-I` curated from the $50,300$ tweets collection using Twitter Streaming API* over four events that happened in India during August-December 2016, namely *Uri Attack, Sur-*

*http://docs.tweepy.org

Table 6.2: Statistical characteristics of the dataset

Heterogeneous Multi-layer Tweet Network

Relation	#Nodes	#Edges*	Edge-type
Hashtag-Hashtag, A^H	3552	10776	Undirected
Mention-Mention, A^M	4243	12277	Undirected
Keyword-Keyword, A^K	28962	181849	Directed
Hashtag-Mention, B^{HM}	6446	13765	Undirected
Hashtag-Keyword, B^{HK}	4782	6648	Directed
Mention-Keyword, B^{MK}	7958	14790	Directed
Keyword-Hashtag, B^{KH}	6824	11825	Directed
Keyword-Mention, B^{KM}	4018	5813	Directed

* The edges are weighted by normalized co-occurrence frequency.

Tweet Corpus

Dataset	#Positive	#Negative	#Neutral	Total Tweets
`Societal-I`	16375	17047	9000	42422

gical Strike, GST Amendment Bill, and Demonetization. Two annotators with strong command on English and Hindi are engaged to annotate the tweets with *positive, negative,* and *neutral* sentiments. We have selected $42,422$ tweets where the two annotators have agreed on the same sentiment, which is of 85% agreement having 82.35 Kappa coefficient scores. The majority of the disagreements among the annotators are on the tweets with stance and sarcastic natures. A similar observation is also reported in[46]. The `Societal` dataset contains 18% non-English tweets (i.e., Hindi and code-mix with English), of which $1,626$ code-mix tweets and $1,505$ tweets with less than five keywords are kept unseen for evaluation of our proposed model. Meanwhile, the hashtags and mentions cover 11% and 15% of the total $39,428$ unique vocabulary of the `Societal` dataset. This dataset is used to build sentiment classifiers and construct a multi-layer network to generate node embeddings. Details of the dataset is shown in Table 6.2.

6.4.2 Embedding method

We investigate the efficacy of our proposed multi-layer network using four different types of node embedding methods namely Multiplex Network Embedding (MNE)[142], Multi-View Embedding (MVE)[102], FastText (FT)[13], and Sentiment Hashtag Embedding (SHE)[113] (listed in Table 6.1). These embedding methods need a collection of node sequences. This study represents the tweet corpus into an expanded multi-layer network by combining the whole tweet networks to generate node sequences via a random walk method. For experimental comparison, we investigate three random walk methods to generate the node sequences, namely *Unbiased* random walk used in MNE, biased random walk used in Node2Vec (*N2V*)[40] and the proposed centrality aware *Biased* random walk. Moreover, to investigate the efficacy of our proposed random walk (RW), we modeled the generated *Biased* RW sequences using the FastText embedding model – which we refer to as *Biased* FT (BFT) in Table 6.3.

6.4.3 Selection of n random walks

A random walker can generate various node sequences starting from a node in the given network. However, all of the sequences are not useful. To identify the node sequences of our interest, we consider a simple second-order Markov chain based language model[56] by calculating the probability of generating a node sequence given a tweet network. This study considers the top three random-walk sequences.*

*We have considered only the top few walks (3, 5, and 7) with the highest probability. Experiments show that considering the top 3 walks provide the best results. The codes for this study are available at: https://github.com/gloitongbam/SA_Hetero_Net

Table 6.3: Performance of sentiment classifiers across different embedding and representations. Blue: Embedding method that performs best for each tweet representations. Red: Best performing tweet representation for each embedding models. Purple: Best performing classifier across different representation of tweet and embedding models. **Purple bold**: Overall best.

Types of tweet representation		Accuracy (in %)								F-Macro (in %)							
		CNN				Bi-LSTM				CNN				Bi-LSTM			
	RW	BFT	MNE	MVE	SHE	BFT	MNE	MVE	SHE	BFT	MNE	MVE	SHE	BFT	MNE	MVE	SHE
Original Tweet	–	77.92	75.53	77.01	76.89	75.22	74.53	73.64	76.05	76.62	73.52	75.33	75.38	72.43	72.59	71.60	74.39
[A] **T+MLN**	Unbiased	73.96	74.90	75.10	76.51	74.83	74.38	73.90	75.70	70.99	72.14	72.68	73.49	71.88	72.62	71.69	72.67
	N2V	75.61	75.45	75.02	74.15	74.65	72.57	72.84	73.84	72.56	73.03	72.83	71.68	72.09	70.51	70.70	70.82
	Biased	77.88	74.30	74.39	77.27	75.89	74.70	74.37	75.63	75.07	71.34	72.83	74.85	73.35	72.58	72.73	73.00
[B] **T+MLN+NE**	Unbiased	76.20	75.30	75.08	77.18	75.31	74.96	74.53	75.51	73.85	72.93	73.04	74.48	72.87	71.63	72.17	73.08
	N2V	75.30	73.80	72.67	73.84	74.54	74.77	72.49	73.84	72.46	71.47	70.91	72.13	72.25	72.50	70.75	71.85
	Biased	78.33	76.57	76.54	77.88	76.33	75.08	75.05	76.53	76.84	74.15	73.01	75.05	74.92	73.41	73.32	74.44
[C] **T+MLN+SNE**	Unbiased	78.72	76.20	77.17	79.37	76.97	74.87	75.73	76.79	77.39	76.43	75.52	78.09	75.73	73.08	74.32	73.84
	N2V	77.77	76.66	77.38	76.87	76.72	72.45	76.47	76.11	76.68	75.50	76.13	74.65	75.30	70.86	73.41	73.69
	Biased	79.23	77.97	78.14	**80.78**	78.95	77.11	78.16	79.33	77.33	76.73	76.90	**79.79**	77.39	75.79	76.66	78.22
[D] **T**+Shuffle	Unfiltered	73.86	76.66	76.26	77.49	74.98	75.05	76.26	76.33	73.04	75.15	74.20	75.04	72.91	73.29	74.54	73.93
	Filtered	77.54	77.17	77.84	77.89	76.21	76.84	76.98	77.78	76.48	75.95	76.43	75.07	75.07	75.32	75.18	76.17

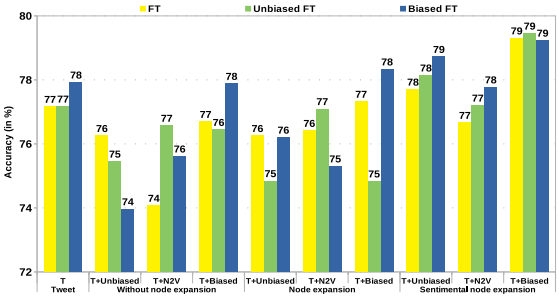
* **T**: Tweet, **MLN**: Multi-layer Network, **NE**: Node Expansion, **SNE**: Sentiment polarized Node Expansion

6.5 Results and observations

In Table 6.3, we show the performance of two sentiment classifiers CNN[88] and Bi-LSTM[136] in terms of accuracy and F-Macro scores over the *Societal* dataset using 10-fold cross validation approach for four embedding models of our choice namely Multiplex Network Embedding (MNE)[142], Multi-View Embedding (MVE)[102], Fast-Text (FT)[13], and Sentiment Hashtag Embedding (SHE)[113]. We consider the work of Nguyen et al.[88] and Xu et al.[136] as the baseline models for text-based sentiment classification of tweet. Along the rows of Table 6.3, we have three groups namely [A], [B] and [C] pertaining to the three types of tweet-network representations, where we compare three different types of Random-Walks (RWs) – *Unbiased*, Node2Vec (*N2V*) and the proposed *Biased* RW to generate node sequences required as inputs for the above embedding methods. From the table, we can see that the network representation of tweets helps the sentiment classification task. Though the tweet-text only classification (in the first row) is hard to beat using the multi-layer network representation of a tweet without node expansion, but for Bi-LSTM based classifier, the classifiers using *Biased* RW in the group [A] beats text only prediction in 75% of the cases with a maximum of 1.13% difference

in terms of F-macro using *Biased* FT embeddings. For CNN classifier, the *Biased* RW in [A] beats original tweet prediction using SHE embeddings. Although the classifiers in [B] gave a competitive performance as compared to text-only classifiers in [A], sentiment polarized node expansion (SNE) method in [C] beats tweet-text based prediction by a margin of 1.4%, and 1.9% (on average) for CNN and Bi-LSTM classifiers respectively – indicating the network representation of tweets, especially when augmented with informative nodes, are useful and complements the text in tweets. Among the RW based methods for node sequence generation, the proposed *Biased* RW performs the best followed by *Unbiased* and *N2V*. The proposed *Biased* RW outperforms *Unbiased* RW decently – can be seen with prominence in [A] *Biased* vs *Unbiased* RWs for CNN classifiers using *Biased* FT embedding. Even the best performances in both the metrics pertain to [C] *Biased* RW with SHE embedding using both the classifiers. We feel the *N2V* style global topology-based biasing is not that useful for sentiment prediction than our biased approach, which uses centrality scores intuitively. Among the embedding models, we observe that *Biased* FT and SHE give competitive performances. We believe *Biased* FT performs competitively as it is trained on centrality-aware random-walks, additionally augmented with sentiment polarized nodes. Whereas, SHE systematically embeds sentiment information and also aided by biased tweet graph view – this makes it an unbeatable performer for sentiment classification.

To realize the importance of generating node sequences with an effective RW method over the proposed network, we investigate another experimental setup by randomly shuffling the selected nodes for expansion (both sentiment polarized and non-polarized nodes) with the tweet text. We call it as T+Shuffle–*Filtered* and *Unfiltered* methods for shuffling of sentiment polarized and non-polarized node



* The plot shows different scale but of same value due to round-off error.

Figure 6.2: Performance of CNN classifier using different types of node embedding generated via FastText algorithm

expansions respectively in [D]. For Bi-LSTM, we can see [D] *Unfiltered* beats text-only prediction, which signifies that the list of selected nodes, though randomly shuffled, but are informative enough. For both the classifiers in [D] *Filtered* outperforms text-only prediction on average by $0.8\%, 2.4\%$, respectively, signifying selected nodes by sentiment polarized node expansion method aids in performance. Here we shall also showcase the novelty of node sequences over a randomly shuffled list of the same nodes. [D] *Unfiltered* is comparable with [B] view – *Biased* RWs are seen to improve upon the prior. Whereas walks in the [C] view, which is comparable to [D] *Filtered* are seen to improve the performance of the latter. [C] *Biased* RW beats [D] *Filtered* by $1.6\%, 1.5\%$ points on average for CNN and Bi-LSTM.

6.5.1 Novelty of centrality-aware walks

It is evident from the already-shown results that our proposed biased random-walks are useful for the effective representation of tweets. One may be further interested in knowing how far these *Biased* RW sequences can improve any embedding models' performance. We conduct a pilot study by creating three versions of the FastText algorithm – a word embedding based original version (FT), an *Unbi-*

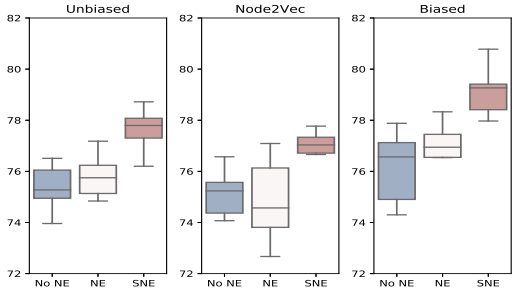
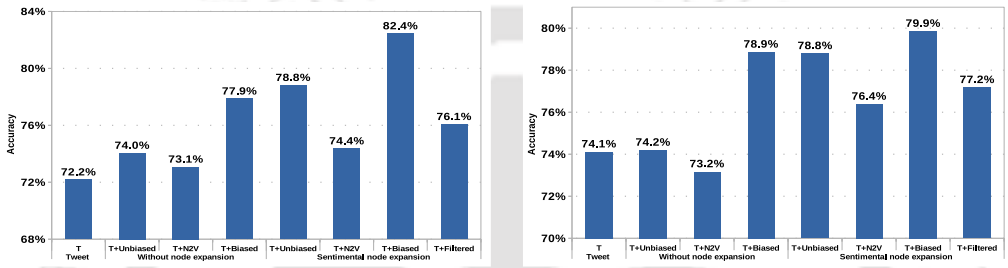


Figure 6.3: Effectiveness of (sentiment polarized) node expansion in tweet-network representations. A:Unbiased, B:Node2Vec, C:Biased representation of tweet-network for No Node Expansion (No NE), Node Expansion (NE), sentiment polarized node expansion (SNE) methods. Accuracy(%) of sentiment prediction is in Y-axis.



(a) Tweets with keywords < 5

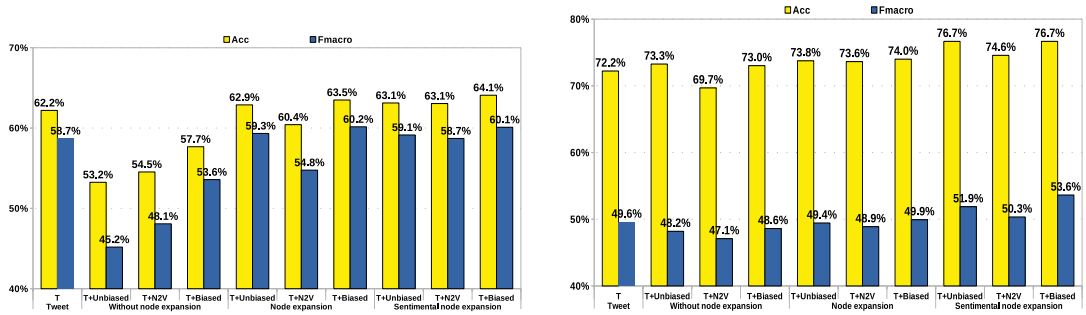
(b) Multilingual tweets

Figure 6.4: Performance of CNN classifier for different under-specified and multi-lingual tweet categories. Inputs to classifier are 5 different tweet representations; i.e. (i) tweet-text only, and node expansion over the actual tweet using random walkers based on (ii) MNE (Unbiased), (iii) Node2Vec (N2V), and (iv) centrality biased node expansions (Biased), and (v) random shuffled of the selected sentiment polarized nodes (Filtered).

ased RW sequence-based version (*Unbiased* FT), and a *Biased* RW sequence-based version (*Biased* FT) as summarized in Figure 6.2. *Biased* FT beats tweet-based FT in 6 out of 10 cases by an average of 1.11%. *Biased* FT also beats *Unbiased* FT in 6/10 cases by an average of 1.37%. Although *Unbiased* FT seems to perform poorer as compared to the original FT in general, in the case of sentiment polarized node expansion, it consistently outperformed the FT – which again proves the effectiveness of the sentiment polarized node expansion method.

6.5.2 Novelty of sentiment polarized node expansion

In this section, we further analyzed the effectiveness of node expansion for the sentiment classification task. We summarize using box-plot in Figure 6.3, the performances of the tweet-network representations (shown in Table 6.3) for sentiment polarized and non-polarized node expansion, and without node expansion over different RW algorithms (i.e. *Unbiased*, Node2Vec, *Biased*). From the figure, it is observed that for each RW methods, the node expansion based representation beats the performance of the tweet representation without any node expansion. Precisely, the sentiment polarized node expansion beats the performance of classifiers with and without non-polarized node expansion by an average margin of 9.19% and 10.57%, respectively. Further, the non-polarized node expansion beats the performance of the classifiers without node expansion by 1.38%. From Figure 6.3, we observe two aspects; – i) the expansion of semantically related nodes in tweet-network makes the performance of centrality based biasing algorithm more reliable, ii) the box-plot of sentiment polarized node expansion methods has a small variance, indicating that it is a pretty stable, reliable method to enhance the tweet network view. Hence we can conclude that extending the networked-view of a tweet by including a few semantically similar, central nodes serves our purpose decently. Further, the performance is enhanced in a considerable margin by adding only the sentiment polarized nodes related to the tweet.



(a) SemEval-2013 (b) SemEval-2016

Figure 6.5: Performance of CNN classifiers across SemEval challenge datasets

6.5.3 Response on under-specified Tweets

We consider tweets having less than five keywords* as an under-specified tweet. Tweets with fewer keywords, although informative, can pose challenges to sentiment classifiers due to under-specificity. We considered the CNN-based classifiers trained using *Biased* FT embedding to classify the under-specified tweets for this study. Figure 6.4(a) shows the CNN-based classifiers' performance based on the different types of tweet representations. From the figure, we observed that the sentiment classifier trained without any node expansion performs better than the classifier trained with tweet-text only. This observation shows the power of optimally selected n random-walk sequences as an alternative representation of tweets. Among no expansion methods, *Biased* RW sequences give the best performance – beat tweet-text only prediction by 5.7% and *Unbiased* RW by 3.82%. We can see similar trends of performance for RW based sequences in case of sentiment polarized node expansion also. However, sentiment polarized node expansion strategically mitigates the problem of under-specified tweets by extending the tweet-network view to include less-noisy informative nodes so that

*Including hashtags and mentions

the generated walks are more diverse and discriminating. The last pair of columns is one special scenario where we give the original tweet-text + list of randomly-shuffled sentiment polarized nodes to the sentiment classifier. This combination (T+*Filtered*) outperforms the tweet only prediction by 3.9% – depicting nodes selected for expansion are important for inference. However, as T+*Biased* without node expansion, T+*Unbiased* and T+*Biased* with sentiment polarized node expansion beat this T+*Filtered* by a margin of 1.8%, 2.7% & 6.4% accuracy respectively. This proves the veracity of this fact that random-walk sequences are a stronger representation of tweets as compared to mere inclusion of a shuffled-list of semantically related words to the tweet-text.

6.5.4 Response on Multilingual tweet

Figure 6.4(b) shows sentiment classification performance over the multilingual tweets – tweet-text written in the code-mixed language. This plot also follows similar trends, as reflected in Figure 6.4(a), but we have two striking observations this time. In the case of multilingual tweets, since the co-occurrence of multilingual words is rare, our proposed node expansion methods are useful to retrieve semantically related co-occurring English words that can aid in inference. We verify the same intuition with this plot. We can see the jump in prediction results for sentiment polarized node expansion for T+*Unbiased*, T+*N2V*, and T+*Biased* over their counterparts in the previous group (without node expansion) with a margin of 4.6%, 3.2% and 0.1% accuracy, respectively. It is interesting to see the huge performance improvement of T+*Biased* without node expansion over tweet only prediction by a margin of 4.75% accuracy – which we believe is due to the power of interpretable, centrality-score aided, optimally biased the RW sequences

of multilingual words.

6.5.5 Evaluation on SemEval datasets

We further investigate the performance of the proposed method with two popular Twitter datasets used in SemEval challenges for sentiment analysis; SemEval-2013* and SemEval-2016†. For this study, we consider the train and test split provided in the datasets. Figure 6.5 (a) and (b) shows the performance of the CNN classifier trained over different types of tweet representation using the SemEval-2013 and SemEval-2016 datasets, respectively. For training the CNN classifier, we use *Biased* FT embeddings trained using the challenge datasets. Our proposed centrality aware-based biased random walker through sentiment polarized node expansion has achieved best performance up to 64% accuracy and 60% F-macro score on SemEval-2013 and up to 77% accuracy and 54% F-macro score for SemEval-2016. Further, on comparing the performance of tweet representation between text-based and network-based without node expansion, it is observed that for both datasets, the representation without node expansion could hardly beat text-based representation in F-macro measure. However, for the SemEval-2016 dataset, our proposed method outperforms text-based representation in both the evaluation measures. We see substantial performance gain for *N2V* RW in both the datasets when augmented with any node expansion. For SemEval-2016, a fascinating thing to observe is – *Unbiased* and *Biased* RW-based sequences almost give a comparable performance in terms of accuracy. However, the *Biased* RW view consistently outperformed the *Unbiased* view in F-macro measure in both

*https://www.cs.york.ac.uk/semeval-2013/task2/

†http://saifmohammad.com/WebPages/StanceDataset.htm

datasets for each of the cases of node expansion. This points to the fact that our method consistently performs better than its counterpart methods.

6.6 Summary

This study investigates the efficacy of transforming tweets to heterogeneous multi-layer network for the sentiment classification task. Our proposed centrality aware random-walk method can generate walk sequences that capture better semantic relations than its unbiased and biased random walk based counterparts. From various experimental observations, it is evident that sentiment-oriented node expansion can reduce under-specificity, noise in a tweet, and enhance the representation. The proposed method outperforms its text-based counterpart in a majority of the cases.

It is observed that representation of tweets to heterogeneous multi-layer networks are helpful for sentiment analysis task. With the recent advancement of the network representation learning approaches, we investigate the use of text and network views for the sentiment classification task from a multi-view learning perspective in the following chapter.

To change ourselves effectively, we first had to change our perceptions.

Stephen R. Covey, American educator

7

Sentiment Analysis of Tweets using Text and Graph Multi-views

Sentiment analysis of tweets is challenging due to data sparsity, under-specificity, noise, and multilingual content. With the recent advances in deep learning frameworks, various studies have attempted to address the above issues through text and network-based representation learning approaches — the text-based approach attempts to capture local semantic and syntactic relations from the sequence of

words. In comparison, the network-based approach tries to capture long-distance semantic relations of the nodes (i.e., non-sequential words) from a network structure. However, limited studies on combining textual and structural (graph) representations of the tweet for the sentiment classification task have been carried out. This study proposes a multi-view learning framework by exploiting both text-based and graph-based representation learning approaches to address the challenges of tweet sentiment classification tasks. To evaluate the efficacy of the proposed framework, this study explores *end-to-end* and *ensemble*-based frameworks for combining both textual and structural views. From various experimental studies, it is observed that combining both views can achieve better performance of sentiment classification tasks than its counterparts.

7.1 Introduction

With the growing popularity of Twitter, tweets have been popularly considered as the target domain for sentiment analysis studies in recent times. Unlike regular text, sentiment analysis on tweets needs to handle some inherent challenges like under-specificity due to limited characters, informal writing styles, misspelling, code-switching, code-mixing contents, etc. Researchers have adopted various approaches such as sentiment-specific representation learning[113,30,119,48], tweet expansion[114,6], users relationship characteristics[148], multi-source information[149], ensembling[5,8,130], etc. to mitigate the above challenges. In the earlier studies, advantages of exploiting network embedding in sentiment analysis of tweets have been reported[114,69,75,141]. It is also reported that network embedding is less sensitive to the social media-related noise mentioned above. Network embedding in senti-

ment analysis has generally been explored from two aspects; *global representation learning* and *local representation learning*. In the studies[114,69], authors construct a global network and learn the representation of required attributes such as keywords, hashtags, users, etc., for sentiment classification. Whereas, studies[75,141] construct a local network of the individual tweet and learn a representation of the tweet for further classification. They represent every tweet using a dependency parse tree to capture structural information and apply GCN[50] with BERT[28] embedding to generate tweet representation. Their observation shows that capturing structural information helps in enhancing sentiment analysis performance.

Motivated by the above observations (i.e., advantages of capturing structural information in the tweet, advantages of using network embedding), this study proposes a multi-view based neural model to exploit both the textual and structural properties for an improved sentiment analysis system and attempt to understand two research questions – (i) *How informative is a graph-based representation of a tweet compared to text-based representation?* (ii) *Does the text-based representation and graph-based representation complement each other?* As in the studies[75,141], using a dependency parse tree to represent a tweet may not always be feasible in the case of multilingual contents (code switch and code mix) and informal textual constructs. While incorporating both structural and textual information is important for sentiment analysis, one needs to consider a graphical method that is insensitive to language and textual construct. Instead of the dependency parse tree, this study proposes to use a heterogeneous multi-layer network to represent a tweet and capture its structural properties. A multi-layer network is a network formed by connecting different layers of networks. For example, a tweet or a collection of tweets can be represented as a heterogeneous multi-layer net-

Tweet:*Historic day for the Nation, #GST bill passed in Lok Sabha. #Congratulations to the nation,salute 2the vision of #PM @narendramodi ji*

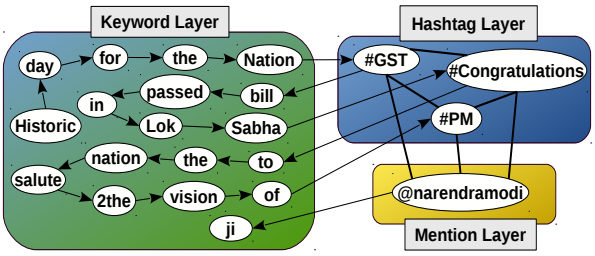


Figure 7.1: An example of representing a tweet to a heterogeneous multi-layer network structure

work by connecting layers of mention's relations, hashtag's relations, keyword's co-occurrence relations. Figure 7.1 shows an example of representing a tweet to a heterogeneous multi-layer network. Since the proposed heterogeneous multi-layer network exploits co-occurrence characteristics rather than the linguistic structure, the heterogeneous multi-layer network is less sensitive to social media-related multilingual noise.

In our proposed method, we generate two views of a tweet, namely *textual view* and *graphical view*. The representation from each view can be generated using an appropriate embedding method. In this study, we use Convolution Neural Network (CNN)[48] and Bidirectional Encoder Representations from Transformers (BERT)[28] based representation learning for textual view and Deep Graph CNN (DGCNN)[145] and Segmented-Graph BERT (Seg-BERT)[143] for graphical view. The representations thus obtained are then integrated using an attention-based aggregator. From various experimental setups over three datasets, it is evident that the proposed multi-view model provides better sentiment analysis performance than its single view counterparts. Further, it is also observed that the proposed model is less sensitive to under-specificity, noise, and multi-lingual

content.

In summary, this chapter has the following contributions:

- Representation of tweet using a language insensitive heterogeneous multi-layer network.
- Evaluate the performance of graph-based representation of tweets compared to its text-based representation.
- Proposed a method to incorporate text and graph views via a multi-view learning framework.

The remainder of the chapter is organized as follows. In Section 7.2, the literature related to this study is presented. Section 7.3 presents the proposed investigation study. The experimental setup is described in Section 7.4. The results and observations are analyzed in Section 7.5. Finally, the study of this chapter concludes in Section 7.6.

7.2 Related studies

There exist a few studies that exploit both text and graph views for sentiment analysis tasks. This section presents a brief review of the literature related to the proposed study. Recent studies have started exploiting graph representation-based methods on top of the text-based representation for sentiment analysis tasks. Studies in[21,141,75] have considered using the Graph Convolution neural Network (GCN) for learning the node features in the aspect-based sentiment classification tasks by transforming the opinionated text to a tree using a dependency parser of the English language. Zhang et al.[141] apply GCN over the dependency tree of

the input text with its node features generated using Long short-term memory (LSTM) model capturing the contextual information of the text. To obtain the aspect-specific features, they apply masking over the GCN output to filter out the non-aspects words features and apply attention over these aspect-aware features for the sentiment classification task. In a similar approach, Meng et al.[75] consider using BERT embedding for learning contextual node features of GCN. Chen et al.[21] perform aspect-based sentiment classification in a multi-view learning framework. This study employs GCN over the dependency tree and LSTM over the word sequence and concatenates the learning representation for the aspect-based sentiment classification task. The difference between the above studies and our proposed study is the application of network representation. The above studies consider using a dependency tree of the input text for aspect-based sentiment classification tasks. However, it is not feasible to have a dependency tree for every language as tweets are highly multilingual. Unlike the above studies, this study exploits the relations of hashtags, mentions, and regular tokens present in tweets as a heterogeneous multi-layer network.

In a different direction but related, Lu et al.[69] consider GCN and BERT to generate the word embeddings. Their study considers vocabulary graphs to generate node embedding using GCN and pre-trained BERT embedding for text-based representation. The two-word embeddings are concatenated to generate the sentence representation via multi-head attention over the input word embeddings for the underlying sentiment classification task. While Yao et al.[137] perform text classification using GCN by representing the text corpus to a heterogeneous network with the document as one type of node and the informative keywords connecting them. This study applies GCN over the single structure, which requires the train-

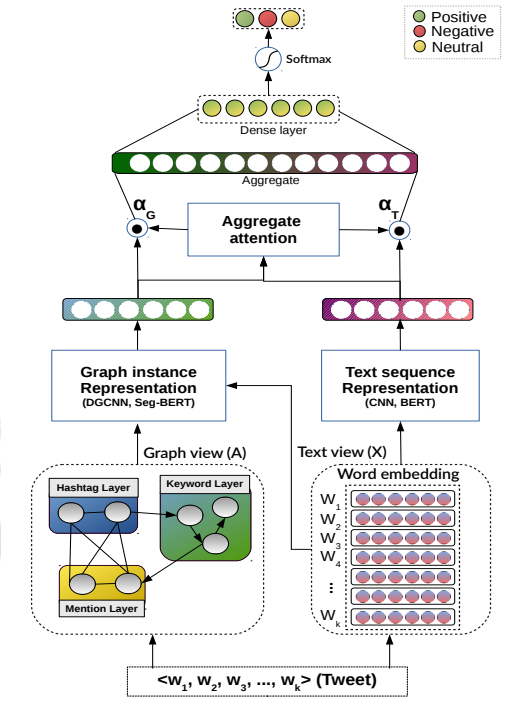


Figure 7.2: Proposed framework for sentiment classification of tweet by incorporating text and graph views through text and graph representation models. $\mathbf{A}$ and $\mathbf{X}$ represent the word embedding and adjacency matrices of the input tweet, and α_i represent the weighted representation of the graph (G) and text (T) representations

ing and testing document to be present in the heterogeneous graph for generating the representation of the document.

7.3 Proposed study

Given a tweet T with n words (w_1, w_2, w_3, ..., w_n), the objective of this study is to incorporate semantic relation of words represented in different views (textual and graph) through a multi-view representation model. The text-view is represented using text embedding methods such as CNN, BERT. The graph-view is represented using graph embedding methods such as DGCNN, Seg-BERT. Figure 7.2 shows a high-level architecture of the proposed framework.

In the remaining part of this section, italic lowercases (e.g., w_i, s), bold lowercases (e.g. $\mathbf{x}_i$, $\mathbf{h}$), and bold uppercases (e.g. $\mathbf{W}$) are used to denote scalars, vectors and matrices respectively. A tweet T is represented in text-view as a matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ where $\mathbf{X}_i$ (i^{th} row of the matrix $\mathbf{X}$) represents the embedding of the word w_i of dimension d. This study considers FastText embedding[13] to generate the initial semantic word embeddings. However, the proposed framework can be applied to any word or node embedding method. The semantics of the word sequence relations are captured using a text representation model F_{seq} that transforms the text-view $\mathbf{X}$ to a vector $\mathbf{z}_{seq}$, i.e., $\mathbf{z}_{seq} = F_{seq}(\mathbf{X}, \theta_{seq})$ where θ_{seq} is the model learning parameter. Since hashtags and mentions are added by the author of the tweet, capturing the relation of hashtags, mentions, and normal tokens will be of great interest as hashtags and mentions can link tweets to similar topics or themes. In order to capture the semantic relations of the words, the tweet T is represented in graph-view as a heterogeneous multi-layer graph via an adjacency matrix representation $\mathbf{A}_{n \times n}$ to accommodate the relation of hashtags, mentions, and normal keywords present in the tweet. The process of representing T to the heterogeneous multi-layer graph is discussed in Section 7.3.2. The semantics of the relations of words are captured using graph instance representation learning model F_{graph} that transformed $\mathbf{A}_{n \times n}$ to a vector $\mathbf{z}_{graph}$ using its corresponding word embedding $\mathbf{X}$ as nodes features, i.e., $\mathbf{z}_{graph} = F_{graph}(\mathbf{A}, \mathbf{X}, \theta_{graph})$ where θ_{graph} is the model learning parameter. This study exploits CNN and BERT models as the text representation model (F_{seq}) for capturing the local semantics of tweets. While DGCNN and Seg-BERT models are considered as the graph representation model (F_{graph}) to capture the semantic relations of the tokens in tweets. The text and graph representation models considered in this study are further discussed in

Sections 7.3.1 and 7.3.2.

Given a text-view representation $\mathbf{z}_{seq}$ and graph-view representation $\mathbf{z}_{graph}$ of a tweet T, the two views are integrated using the *Scaled Dot-Product Attention* mechanism[127]. Given a query tweet, the idea is to assign attention weights to text-view and graph-view. We define the query of the attention by element-wise average of the $\mathbf{z}_{seq}$ and $\mathbf{z}_{graph}$ representations, i.e.,

$$\mathbf{z}_{avg}[i] = \frac{\mathbf{z}_{seq}[i] + \mathbf{z}_{graph}[i]}{2} \tag{7.1}$$

The attention weight vector of the text view is defined as:

$$_{seq} = Softmax(\frac{\mathbf{z}_{avg} \cdot \mathbf{z}_{seq}^{T}}{\sqrt{|\mathbf{z}_{avg}|}}) \tag{7.2}$$

Similarly, the attention weight vector of the graph view is defined as:

$$_{graph} = Softmax(\frac{\mathbf{z}_{avg} \cdot \mathbf{z}_{graph}^{T}}{\sqrt{|\mathbf{z}_{avg}|}}) \tag{7.3}$$

Now, the two views are integrated by concatenating the weighted representation of each views as follow:

$$\mathbf{z}_{agg} = \alpha_{seq} \cdot \mathbf{z}_{seq} \oplus \alpha_{graph} \cdot \mathbf{z}_{graph} \tag{7.4}$$

The classifier is built using a dense layer with Relu activation function. The classification output $\mathbf{s}$ is define as:

$$\mathbf{s} = Softmax(Relu(\mathbf{W} \cdot \mathbf{z}_{agg} + \mathbf{b})) \tag{7.5}$$

where $\mathbf{W}$ and $\mathbf{b}$ and weight and bias parameters of the dense layer. We use *Categorical Cross-Entropy* loss function define in Equation 7.6 and *Adam Optimizer*[49] as the optimization technique for training the propose framework.

$$\Delta = -\frac{1}{l}\sum_{i=1}^{l}\sum_{c}\mathbf{t}_{ic}log(s_{ic}) \tag{7.6}$$

where c is the number of sentiment classes, $\mathbf{t}_{ic}$ is the c^{th} ground truth class for the tweet, l is the total number of training samples, and $\mathbf{s}_{ic}$ is the predicted probability on sample i for the c^{th} class.

7.3.1 Text representation model

Given a tweet, text-view can be generated using any suitable text embedding methods. In this study, we have investigated text representation using CNN[48] and BERT[28]. This section discusses CNN and BERT based embedding briefly.

Convolution Neural Network

From various studies[114,115,57], it is reported that CNN captures local semantics specially for short text better than recurrent-based models for sentiment classification tasks. If a matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$ defines a tweet, then, the i^{th} row of the matrix $\mathbf{X}$ represents the embedding of the i^{th} word in the tweet. To apply convolution over the matrix $\mathbf{X}$, we consider kernels of size $h \times d$ to capture spatial properties of h consecutive words in the tweet. We apply a filter f at a position t of $\mathbf{X}$ using the following expression.

$$conv_t^{(f)}(\mathbf{X}, h) = ReLu(\mathbf{W}^{(f)} \cdot \mathbf{X}_{t:t+h-1} + b^{(f)}) \tag{7.7}$$

where $\mathbf{W}^{(f)}$ is the kernel matrix for the filter f and $b^{(f)}$ is the corresponding bias. We consider padding and apply filter f with a stride size 1 to obtain a convolution vector $\mathbf{c}^{(f)}$ for the tweet matrix $\mathbf{X}$. The elements of $\mathbf{c}^{(f)}$ vector are defined as follow:

$$\mathbf{c}_i^{(f)} = conv_i^{(f)}(\mathbf{X}, b) \tag{7.8}$$

After applying *maxpooling*, we obtain a vector $\mathbf{z}^f$ to represent the tweet using the filter f i.e.,

$$\mathbf{z}^{(f)} = maxpooling(\mathbf{c}^{(f)}) \tag{7.9}$$

We consider 128 number of filters. The 128 $\mathbf{z}^f$ vectors obtained from 128 filters are concatenated to obtain the vector representation of the textual view of the tweet represented by $\mathbf{X}$

$$\mathbf{z} = \mathbf{z}^1 \oplus \mathbf{z}^2 \oplus ... \oplus \mathbf{z}^{128} \tag{7.10}$$

For ease of reference, we can define the whole operation as:

$$\mathbf{z} = CNN(\mathbf{X}, \theta) \tag{7.11}$$

where θ denotes the required hyper-parameters of the CNN model such as k filters, h *convolution* window size. We apply 2-layers of CNN model with same parameters over the input $\mathbf{X}$ to represent the input tweet, i.e.,

$$\mathbf{z}_{cnn} = CNN(CNN(\mathbf{X}, \theta), \theta) \tag{7.12}$$

In order to reduce the size of $\mathbf{z}_{cnn}$ vector, the $\mathbf{z}^i$ vector is further transformed to scalar by applying *global maxpooling* over $\mathbf{z}^i$.

Bidirectional Encoder Representations from Transformers

Majority of the recent study on text embedding considers using Bidirectional Encoder Representations from Transformers, more commonly known as BERT[28]. Earlier studies have considered using BERT as a pre-trained model[75,69]. However, it is inefficient to use a pre-trained BERT model if it does not match the current domain of interest leading to out-of-vocabulary issues[86]. This study considers building BERT from scratch to overcome the inefficiency caused by using pre-trained BERT models.

Given a tweet representation $\mathbf{X} \in \mathbb{R}^{n \times d}$, the BERT model captures the semantic information of the word sequences by relying only on the attention-weighted representation of the words. The word order relation is incorporated into the initial word embedding $\mathbf{X}$ by adding element-wise positional embedding. The position embedding for each word position *pos* can be defined as:

$$\mathbf{P}_{pos,i} = \begin{cases} sin(pos/10000^{2i/d}) & \texttt{if } i \in (1,d) \texttt{ is even} \\ cos(pos/10000^{2i/d}) & \texttt{otherwise} \end{cases} \tag{7.13}$$

There are l number of transformer blocks stacked on top of the other in the BERT architecture. The initial input to the first transformer block is the sum of word embedding $\mathbf{X}$ and positional embedding $\mathbf{P}$, i.e., $\mathbf{Z}_0 = \mathbf{X} + \mathbf{P}$. To capture the different aspects of tweet semantics, a transformer block t can have mh multi-head attention layers. For each attention head $i \in (1, mh)$ in a transformer block t, three matrices are generated using dense layer over the input $\mathbf{Z}_t$ serving as the *query*, *key*, and *value* to find the attention-weighted representation using the

Scaled Dot-Product Attention mechanism[127], i.e.,

$$\begin{aligned} \mathbf{Q}_i &= \mathbf{W}_{qi} \cdot \mathbf{Z}_t \\ \mathbf{K}_i &= \mathbf{W}_{ki} \cdot \mathbf{Z}_t \\ \mathbf{V}_i &= \mathbf{W}_{vi} \cdot \mathbf{Z}_t \end{aligned} \tag{7.14}$$

where $\mathbf{Q}_i, \mathbf{K}_i$, and $\mathbf{V}_i$ is a linear transformation of the input $\mathbf{Z}_t$ through three different weight parameters $\{\mathbf{W}_{qi}, \mathbf{W}_{ki}$, and $\mathbf{W}_{vi}\} \in \mathbb{R}^{n \times n}$. The output of each attention head $i \in (1, mh)$ in a transformer block t can be defined as:

$$\mathbf{Y}_t^{(i)} = Softmax(\frac{\mathbf{Q}_t^{(i)} \cdot \mathbf{K}_t^{T(i)}}{\sqrt{|\mathbf{Q}_t^{(i)}|}})\mathbf{V}_t^{(i)} \tag{7.15}$$

The attention-weighted outputs of the multi-head attention layer are concatenated to generate the semantic representation using dense layer with *Relu* activation function as output for the transformer block t, i.e.,

$$\mathbf{Z}_{t+1} = Relu(\mathbf{W} \cdot \mathbf{Y}_{1:mh} + \mathbf{B}) \tag{7.16}$$

where $\mathbf{W} \in \mathbb{R}^{n \times n \cdot mh}$ and $\mathbf{B} \in \mathbb{R}^{n \times d}$ are the weights and biased parameter matrices, $\mathbf{Z}_{t+1}$ represent the output of the t transformer block. The output of the last transformer block, i.e., $\mathbf{Z}_{l+1}$ is considered as the final representation of the input tweet T to the BERT model. To represent in the vector space, $\mathbf{Z}_{l+1}$ is being flatten into $\mathbf{z}_{bert} \in \mathbb{R}^{n \cdot d \times 1}$ vector for sentiment classification. For ease of reference, the whole operation can be defined as:

$$\mathbf{z}_{bert} = BERT(\mathbf{Z}_0, \theta) \tag{7.17}$$

where θ represents the hyper-parameters such as l number of encoders, mh number of multi-head attentions, d hidden layer dimensions. We have considered the same hyper parameters used in original BERT setup, i.e., $l = 8$ transformer blocks and $mh = 8$ multi-head attentions.

7.3.2 Tweet graph construction

A tweet can be represented as a heterogeneous multi-layer network by considering the relation of hashtags, mentions, and normal tokens co-occurring in a tweet. This study consider three types of undirected co-occurring relations i.e. mention–mention (MM), hashtag–hashtag (HH), mention–hashtag (MH) or hashtag-mention (HM) and five directed relations i.e. keyword $\rightarrow$ keyword (KK), keyword $\rightarrow$ hashtag (KH), hashtag $\rightarrow$ keyword (HK), keyword $\rightarrow$ mention (KM), and mention $\rightarrow$ keyword (MK) to represent a tweet in a heterogeneous multi-layer network. The directed edges are considered to capture the sequence relation of normal tokens. Figure 7.1 shows an example of how a tweet is represented in the heterogeneous multi-layer network. Accommodating all the eight types of relations of a tweet with n tokens can be represented in the adjacency matrix as:

$$\mathbf{A}_{n\times n} = \begin{bmatrix} \mathbf{B}^{HH} & \mathbf{B}^{HM} & \mathbf{B}^{HK} \\ \mathbf{B}^{MH} & \mathbf{B}^{MM} & \mathbf{B}^{MK} \\ \mathbf{B}^{KH} & \mathbf{B}^{KM} & \mathbf{B}^{KK} \end{bmatrix} \tag{7.18}$$

where $MH = HM$ and $\mathbf{B}^r$ represent the adjacency matrix representation of the relation $r \in \{HH, MH, HM, MM, KK, KH, HK, KM, MK\}$.

Network expansion: This study investigates *whether adding semantically re-*

lated tokens into the tweet-graph can enrich the representation of the tweet. To expand a tweet graph, the semantically related nodes of all tokens in the tweet are retrieved using cosine similarity over the word embeddings generated using FastText (FT) and Sentiment Hashtag Embedding (SHE) methods. We select top 20 tokens having high cosine similarity scores to the tokens present in the tweet as semantically relevant nodes of the tweet. These 20 nodes are added to the tweet graph by introducing an undirected edge with all the nodes. For ease of reference, such node expansion approach is considered as semantic *Node Expansion* (`NE`).

Further, we investigate *whether adding semantically related as well as sentiment polarized tokens into the tweet-graph can enrich the representation of the tweet or not.* For this study, the previously selected semantically similar nodes through `NE` is filtered by selecting only the sentiment polarized tokens. To select the sentiment polarized tokens, this study exploits the SHE method to classify the sentiment of the 20 semantically relevant nodes. Then, the sentiment polarized node expansion is performed by dividing the 20 nodes into three different sentiment sets, i.e., positive, negative, and neutral. The dominating sentiment set, i.e., majority of the nodes having same sentiment, are selected for sentiment polarized node expansion. For ease of reference, this study consider such expansion approach as *Sentiment polarized Node Expansion* (`SNE`).

7.3.3 Graph representation model

Recent studies on graph instance representation learning[143,145] have shown a promising results in capturing the latent representation of the graph. We can apply graph instance representation learning methods such as DGCNN[145] and Seg-BERT[143] over $\mathbf{A}_{n\times n}$ to represent it in vector space for graph classification

task.

Deep Graph Convolution Neural Network

Zhang et al.[145] have used Graph Convolution Neural network (GCN)[50] for graph classification task. Compared to the study of Kipf and Welling[50] which work on single structure, this method is able to represent graphs of arbitrary structures. They proposed an algorithm named *SortPooling* similar to Weisfeiler-Lehman node coloring algorithm[58] for sorting vertex features to learn the global graph topology.

Given a graph $\mathbf{A}_{n\times n}$ and feature matrix (word embedding) $\mathbf{X} \in \mathbb{R}^{n\times d}$, we can apply multiple stacks of *GCN* at time t to output $\mathbf{Z}^t$ as

$$GCN(\mathbf{Z}^{t-1}, \mathbf{A}) = ReLu(\tilde{\mathbf{A}}\mathbf{Z}^{t-1}\mathbf{W})$$

where $\tilde{\mathbf{A}} \in \mathbb{R}^{n\times n}$ is the adjacency matrix with added self-loops (identity matrix) i.e. $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, $\mathbf{Z}^0 = \mathbf{X}$, $\mathbf{W} \in \mathbb{R}^{d\times c}$ is the neural weight parameters* shared with all the graphs, and h is the number of *GCN* layers. For learning global node features, the output of each *GCN* layers are concatenated row-wise i.e. $\mathbf{Z} = \mathbf{Z}^{1:h}$ and apply *SortPooling* over $\mathbf{Z}$ i.e., $\mathbf{Z}_{sp} = SortPooling(\mathbf{Z})$. The output $\mathbf{Z}_{sp}$ is fed to *CNN* layer to generate the graph representation via *MaxPooling*, i.e.,

$$\mathbf{z}_{dgcn} = MaxPool(CNN(Z_{sp}, \theta_{graph}))$$

where θ_{graph} is the learning parameters of CNN. We use the same parameters considered in text representation model (refer Section 7.3.1).

*For generalization we set $c = d$

Segmented-Graph BERT

Zhang et al.[143] have used BERT architecture to encode graph information given node features such as word embeddings ($\mathbf{X}$), latent representation of adjacency neighborhood matrix ($\mathbf{A}$), node degree matrix ($\mathbf{D}$), and node global role matrix ($\mathbf{WL}$) pre-computed using Weisfeiler-Lehman algorithm[58]. We feed these features as input to the BERT model, i.e.,

$$\mathbf{Z}_0 = \mathbf{X} + \mathbf{A} + \mathbf{D} + \mathbf{WL} \tag{7.19}$$

Hence, we can learn graph instance representation of a graph similar to normal BERT model which captures semantic relations of the nodes in the graph as

$$\mathbf{z}_{segbert} = BERT(\mathbf{Z}_0, \theta_{graph}) \tag{7.20}$$

where θ_{graph} is the learning parameters of BERT. We use the same parameters considered in text representation model (refer Section 7.3.1).

7.3.4 Multi-view model

The text-based and graph-based representations generated using the above methods are concatenated for tweet classification task. These multi-view representations can be incorporated for classification task either on an *end-to-end* framework or as *ensemble* of the individually trained representations methods. In the *end-to-end* framework both the text and graph representation methods are trained together for the tweet classification task. While in *ensemble* framework, the text and graph representation methods are trained separately. The individual repre-

Table 7.1: Characteristics of the experimental datasets

Dataset	Pos	Neg	Neu	Total	Topics	Domain
Soceital	16,375	17,047	9,000	42,422	*Kashmir Unrest, Pathankot Attack, Surgical Strike, GSTN*	Social Issue
SemEval-2016	1,296	2,491	276	4,063	Atheism, Climate Change, Feminist Movement, Hillary Clinton, Legalization of Abortion	Social Issue
SemEval-2013	5,115	2,017	6,099	13,231	General Discussion	–

sentation generated from both the methods are concatenated together as tweet representation for training ensemble classifier.

7.4 Experimental setup

7.4.1 Dataset

To evaluate the efficacy of the proposed framework, this study considers a *Societal* dataset used in[115,114] for sentiment classification task. This dataset contains 1,505 under-specified tweets (tweets having less than 5 tokens) and 1,626 multilingual tweets (code-mix of Hindi and English languages). The *Societal* dataset is curated over 4 topics happened in India namely Kashmir Unrest, Pathankot Attack, Surgical Strike, and GSTN*. Table 7.1 shows the characteristics of the training dataset considered in this study.

7.4.2 Baseline classifiers

To evaluate the performance of the proposed framework, we consider four single-view classifiers i.e.; *CNN*, *BERT*, *DGCNN*, and *Seg-BERT*, and two multi-view classifiers i.e., *T+MLN* and *VGCN-BERT* as baseline models for comparision.

- **CNN**: The output of $\mathbf{z}_{cnn}$ of CNN model over the input $\mathbf{X}$ is considered as

*https://en.wikipedia.org/wiki/Goods_and_Services_Tax_(India)

Table 7.2: Performance of sentiment classifiers over the *Societal* dataset.

Single-view methods	*Societal*		SemEval-2016		SemEval-2013	
	Accuracy	**F-Macro**	**Accuracy**	**F-Macro**	**Accuracy**	**F-Macro**
T (CNN)	77.16	76.08	73.41	47.26	**64.42**	**61.98**
T (BERT)	76.92	**75.78**	68.59	37.20	55.72	48.20
DGCNN	74.89	72.27	**74.31**	**48.38**	62.03	56.59
Seg-BERT	**77.39**	75.71	70.33	41.48	56.60	51.32
Multi-view methods	*Societal*		SemEval-2016		SemEval-2013	
	Accuracy	**F-Macro**	**Accuracy**	**F-Macro**	**Accuracy**	**F-Macro**
CNN+DGCNN (End-to-end)	78.70	76.83	74.31	47.92	64.84	62.19
CNN+DGCNN (Ensemble)	**79.34**	**77.03**	**74.55**	48.78	**66.00**	**62.79**
BERT+Seg-BERT (End-to-end)	73.36	72.12	68.81	52.26	60.42	54.76
BERT+Seg-BERT (Ensemble)	75.37	73.81	70.11	52.63	62.20	58.71
T+MLN (CNN)	76.69	73.97	72.22	**53.63**	63.49	60.16

the tweet representation for sentiment classification task in Equation 7.5.

- **BERT**: The output of $\mathbf{z}_{bert}$ of BERT model over the input $\mathbf{X}$ is considered as the tweet representation for sentiment classification task in Equation 7.5.
- **DGCNN**: The output of $\mathbf{z}_{dgcnn}$ of DGCNN model over the input $\mathbf{X}$ is considered as the tweet representation for sentiment classification task in Equation 7.5.
- **Seg-BERT**: The output of $\mathbf{z}_{seg-bert}$ of Seg-BERT model over the input $\mathbf{X}$ is considered as the tweet representation for sentiment classification task in Equation 7.5.
- **T+MLN**: Our earlier work[114] is considered as one of the baseline method for incorporating graph as well as text information.

7.5 Results and Observation

In this section, we investigate the efficacy of the proposed framework over the baseline methods through the two research questions – (i) *How informative is a*

graph-based representation of a tweet as compared to that of the text-based representation? (ii) *Does the text-based and graph-based representations complement each other?* The efficacy of the proposed framework is investigated over the *Societal* dataset using a 10-fold cross-validation strategy. Table 7.2 shows the performance of the classifiers over *Societal* and *SemEval* datasets for the sentiment classification task. For this analysis study, the under-specified and multilingual tweets are excluded from the *Societal* dataset. These tweets are considered to investigate whether the proposed model is able to address the challenge of social media noises.

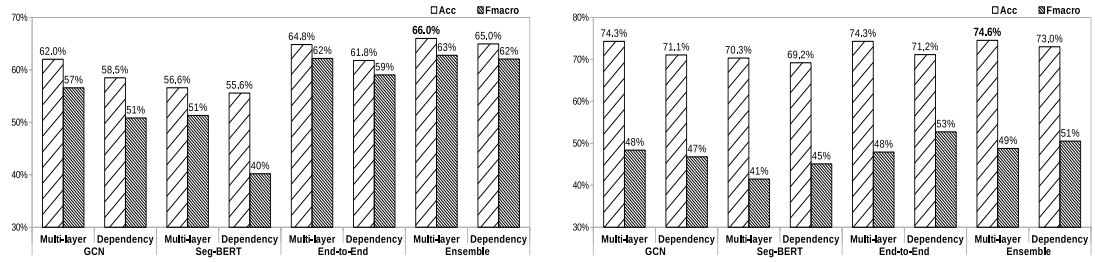
7.5.1 How informative is a graph-based representation of a tweet compared to text-based representation?

The first part of Table 7.2, i.e., single view methods, shows the performances of the single-view classifiers. In the societal dataset, it is observed that the best performance achieved by a single-view classifier is up to 77.39% accuracy with F-Macro of 75.71% using Seg-BERT over the heterogeneous tweet graph. While the performance of the sentiment classifier built over text representation, i.e., text-view, can achieve the best performance up to 77.16% accuracy using CNN. Similarly, in the SemEval 2016 dataset, it is observed that the graph-based classifier DGCNN can achieve best up to 74.31% with 48.38% F-Macro score while the text-based classifier CNN can achieve a comparable performance accuracy of 73.41% and 47.26% F-Macro score over the same dataset. In contrast, it is observed that the CNN classifier can achieve the best performance of 64.42% accuracy with an F-macro score of 61.98% over the SemEval 2013 dataset while the DGCNN classifier can achieve a best up to 62.03% with a 56.59% F-Macro score over the SemEval 2013

dataset. This study shows that the graph-based representation could be effectively used for tweet classification tasks, and it is as informative as the text-based representation.

7.5.2 Does the text-based and graph-based representations complement each other?

It is evident from the second part of Table 7.2, i.e., multi-view methods, that incorporating both the text and graph views have significantly improved the performance of sentiment classifiers for both *end-to-end* and *ensemble* than the single-view based methods. The *ensemble* frameworks using CNN and DGCNN methods can achieve the best performance of up to 79.34% accuracy and 77.03% F-Macro scores. In contrast, its *end-to-end* framework can achieve up to 78.70% accuracy and 76.35% F-Macro score. It is observed that the performance of multi-view classifiers using BERT and Seg-BERT could not improve the performance compared to its individual classifier performances over *Societal* and SemEval 2016 datasets. One of the reasons for not performing well compared to the individual view is that Seg-BERT takes both the text and graph information while encoding graph representation. In contrast, BERT takes only the text information to encode sequence representation. Hence, the tweet representation generated using Seg-BERT has redundant information. Adding BERT information in the multi-view framework has created a noisy representation of the tweet due to the losses while training the multi-view framework. Among the baseline methods for incorporating multi-views, the T+MLN classifier is able to achieve best up to 76.69% accuracy and 73.97% F-macro score. It is also observed that the best performance of the single-view and multi-view classifiers over the SemEval-2016 dataset is relatively comparable.



(a) SemEval-2013 (Task 2-B)

(b) SemEval-2016 (Task 6)

Figure 7.3: Performance of classifiers over SemEval 2013 and 2016 challenge datasets. End-to-end and Ensemble classifiers are combination of CNN and DGCNN methods.

However, a clear difference between single-view and multi-view classifiers' best performance is observed in the Societal and SemEval-2013 dataset. One of the reasons for underperforming is due to the small size corpus. Compared to the *Societal* and SemEval-2013 datasets, the corpus size of the SemEval-2016 dataset is minimal; therefore, the node information in the tweet graphs of this dataset is not fully incorporated. With a larger corpus, the graph representation learning method is able to benefit the global properties of the nodes. As a result, the performance of the *end-to-end* and *ensemble*-based classifiers have significantly improved using DGCNN. This study shows that the node's properties in the tweet graph can inherently be captured with a larger corpus. Further, this study shows that incorporating text and graph views can better enrich the tweet representation for sentiment classification tasks than individual classifier performance. Therefore, from the above investigation, it is evident that both text and graph views complement the representation of the tweet for sentiment classification.

7.5.3 Heterogeneous multi-layer network v/s Dependency tree

This section investigates *if there is a need for a language-dependent dependency parser to construct the tweet graph.* For this study, we consider an off-the-shelf

dependency parser in English language* to construct the tweet graph. Since SemEval datasets are English language datasets, we consider these datasets for the experimental study. The sentiment classification performance of tweets is evaluated over two variant representations of the tweet, i.e., tweet represented using the dependency parser and the heterogeneous multi-layer network. Figure 7.3 shows a performance comparison of the single-view and multi-view classifiers over the tweet graph using dependency parser and the heterogeneous graph. It is evident from the figures that the performances of the single-view classifiers, i.e., DGCNN and Seg-BERT, over the tweet representation using the heterogeneous graph have better classification accuracy than using the dependency graph. It is observed that the best performing classifiers (i.e., *ensemble* classifier) for both the graph representations are relatively comparable. The *ensemble* classifier trained over the SemEval 2013 dataset using the heterogeneous multi-layer network can achieve the best of up to 66% accuracy while using the dependency graph can achieve up to 65% accuracy. Similarly, the *ensemble* classifiers trained over the SemEval 2016 dataset using the heterogeneous multi-layer network can achieve the best of up to 75% accuracy while using the dependency graph can achieve up to 73% accuracy. This study shows that the heterogeneous multi-layer network is language invariant and able to perform better than language-dependent word graph structure.

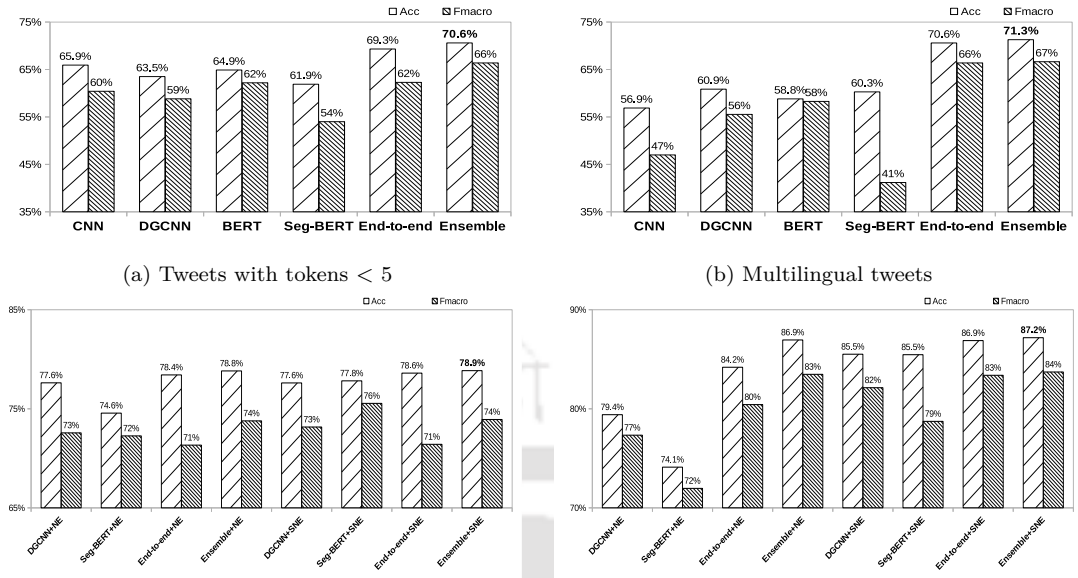
Further, *to investigate whether the heterogeneous multi-layer graph is less sensitive to social media-related noises*, we investigate the performance of the proposed framework over the under-specified and multilingual tweets in the following subsections. In this study, we consider tweets having less than five tokens as under-

*https://spacy.io/usage/linguistic-features

specified tweets. To investigate the performance of the proposed framework over these tweets, the best performing *ensemble*-based classifier (as observed in Table 7.2), i.e., the ensemble of CNN and DGCNN classifiers and the single-view classifiers, are considered. The performances of the single-view classifiers are compared with the *end-to-end* and *ensemble* frameworks using the under-specified and multilingual tweets. Furthermore, we investigate the performance of sentiment classifiers on classifying tweets by adding semantically relevant tokens and sentiment polarized tokens to the tweet-graph through `NE` and `SNE` approaches. For this study, the classifiers are not re-trained over the expanded tweet-graphs. Instead, the representation of tweets is generated from the expanded graph for comparison. To ease of reference, we use the notation *classifier+NE* to indicate the classifier uses the expanded graph generate using either NE or SNE approaches for sentiment classification.

7.5.4 Performance of sentiment classification over under-specified tweets

As mentioned above, to investigate the performance of the proposed framework over the under-specified tweets, the best performing *ensemble*-based classifier (in Table 7.2), i.e., the ensemble of CNN and DGCNN classifiers and the single-view classifiers, i.e., CNN, BERT, DGCNN, Seg-BERT are considered for comparison. From the Figure 7.4(a), it is observed that the proposed framework using *ensemble*-based method outperforms the individual-view-based classifiers by achieving best accuracy up to 70.60% and F-macro score of 66.40%. While, the *end-to-end* framework is able to achieve up to 69.32% accuracy and 62.30% F-macro score. The best performance of a single-view classifier is using CNN classifier, which can



(a) Tweets with tokens < 5

(b) Multilingual tweets

(c) Under-specified tweets after node expansion

(d) Multilingual tweets after node expansion

Figure 7.4: Performance of classifiers over different under-specified and multi-lingual tweet categories. End-to-end and Ensemble classifiers are combination of CNN and DGCNN methods; *classifier*+NE is the *classifier* performance of tweet classification over node expansion graph; *classifier*+SNE is the *classifier* performance of tweet classification over sentiment polarized node expansion graph

achieve up to 65.93% accuracy and 60.40% F-macro score followed by BERT with 64.89% and 62.19% accuracy and F-macro scores respectively. Among the graph based approach, DGCNN can achieve up to 63.51% accuracy and 58.80% F-macro score while Seg-BERT achieves upto 61.89% and 53.99% accuracy and F-macro scores respectively. From this study, it shows that incorporating both views can better represent tweets than representations of their individual views.

Further, after performing semantic *Node Expansion* (`NE`) over the under-specified tweet graph, it is evident from the Figure 7.4(c) that the performance of the classifiers significantly improves. It is observed that with `NE`, the performance of DGCNN+NE and Seg-BERT+NE improves to 77.62% and 74.56% accuracies respectively. Incorporating the text and graph-views using `NE` in the *end-to-end* framework (i.e. *end-to-end*+NE) further improves the classifier performance by

78.42% accuracy. With *ensemble* framework (i.e. *ensemble*+NE), the classifier performance improves up to 78.82% accuracy. Furthermore, after performing *Sentiment polarized Node Expansion* (SNE), the best performance we are able to achieve is up to 78.85% accuracy using the ensemble classifier i.e., Ensemble+SNE. It shows that adding semantically related polarized sentiment nodes in the tweet graph can further enrich the tweet representation even without re-training the classifiers. From this study, it is evident that the proposed framework can address the problem of the under-specificity of tweets with a high margin compared to the performance of the single-view classifiers.

7.5.5 Performance of sentiment classification over multilingual tweets

In the same fashion, as discussed above, this section investigates the performance of the proposed framework over multilingual tweets. It is observed from Figure 7.4(b) that incorporating both the text and graph views in the *ensemble* framework can achieve up to 71.28% accuracy and 66.64% F-macro score. While, the *end-to-end*-based classifier can achieve up to 70.59% accuracy and 66.38% F-macro. Among the single-view classifiers, the DGCNN classifier has achieved the highest of 60.86% accuracy and 55.54% F-macro, followed by Seg-BERT with 60.28% 41.18% accuracy and F-macro scores, respectively. The BERT classifier can achieve up to 59% accuracy and 58% F-macro scores, while the CNN classifier can achieve up to 57% accuracy and 47% F-macro score. This shows that incorporating both text and graph views can better represent a tweet than representing its individual views.

Further, with node expansion of the tweet-graph, the improvement of the per-

formance of classifiers is evident in Figure 7.4(d). The DGCNN and Seg-BERT classifier over the `NE` of tweet-graph can achieve up to 79.41% and 74.1% accuracies respectively. Further, incorporating text representation over the expanded graph using *end-to-end* and *ensemble* frameworks improves the performance of the classifiers by achieving up to 84.19% and 86.95% accuracy respectively. Furthermore, with `SNE` of the tweet-graph, the best performance we are able to achieve is up to 87.17% accuracy using *ensemble* of text representation and 86.95% *end-to-end* representation of `SNE` of tweet-graph. This study shows that the proposed framework of incorporating text and graph views can better enrich the tweet representation than its single-view representation. The tweet representation can be enriched further more by adding semantically related sentiment polarized nodes in the tweet graph. It is also evident that the proposed framework is able to address the problem of multilingual tweets by incorporating both text and graph-views in the multi-view learning framework.

7.6 Summary

This study proposes a multi-view learning framework for sentiment classification of tweets to address under-specificity, noise, and multilingual content by representing tweets using text and graph representation learning methods. To incorporate both text and graph-views in the multi-view learning framework, this study explores both *end-to-end* and *ensemble* based classifiers. It is observed from various experimental studies that the performance of the tweet sentiment classifier improves significantly after incorporating both text and graph views than its individual-view classifiers. The *ensemble* based classifier is able to perform better than *end-to-*

end based classifier on incorporating both the views. Further, it is observed that the proposed framework can perform better than its counterpart in addressing multilingual and under-specified tweets. Moreover, after performing node expansion over the tweet graph, the performance of the classifiers improves furthermore through semantic (`NE`) and sentiment polarized node expansion (`SNE`).

Limitation: Even though the performance of the classifiers improves with `SNE`, it is observed from Figure 7.4(c, d) that the performances of the classifiers over the `NE` and `SNE` of tweet-graphs are relatively comparable. The reason for the less effectiveness of `SNE` could be due to filtering the sentiment polarized nodes. As the sentiment polarized nodes are chosen based only on the dominating sentiment, the number of nodes selected using `SNE` and the total nodes selected using `NE` is almost similar in size. Hence, mechanizing a method to retrieve more relevant sentiment polarized nodes of the input tweet could further enhance the performance of the sentiment classification task.

8 Conclusion

This thesis work investigates the problem of sentiment analysis of tweets on societal topics. With the increase in the availability of public opinions on Twitter, sentiment analysis of tweets has become challenging due to data sparsity, under-specificity, noise, and multilingual content. This thesis addressed the importance and challenges of building sentiment classifiers for the societal domain. In particular, this thesis exploits text and network representation learning techniques to address the challenges of data sparsity, under-specificity, noise, and multilingual

content. In this chapter, the summary of the contributions made in this thesis for sentiment analysis of tweets on societal topics is presented.

8.1 Summary of Thesis

Sentiment analysis is a domain-dependent problem, and most of the sentiment analysis tools are built for customer reviews. Hence, the suitability of using such existing off-the-the-shelf tools for a societal topic is subject to investigation. This thesis first aimed to investigate the performance of the existing off-the-shelf sentiment analysis tools over the tweets in the societal domain. The first contribution of the thesis performs an empirical study to evaluate the performance of 10 publicly available sentiment analysis `Tools` and 17 state-of-the-art machines learning `Techniques` over eight datasets covering various topics of societal, customer reviews, and general discussions. It is evident from various experimental observations that most of the off-the-shelf `Tools` are not suitable for societal topics. However, these tools have shown encouraging performance for customer reviews. From the evaluation of the `Techniques`, we observe neural network-based classifiers dominate feature-based classifiers. We also note that tweets under societal issues collected from different geographical regions share common sentiment characteristics.

People often use hashtags while posting their opinions reflecting meta-information such as sentiment, emotion, topic, and entity of a tweet. As people are free to choose or generate hashtags without much restriction, tweets comprising hashtags experience out-of-vocabulary issues. Since hashtags represent meta information of the tweets, normalizing out-of-vocabulary hashtags to semantically similar ex-

isting hashtags can enhance the sentiment classifier's performance. Further, the sentiment representation of the tokens can better enhance the sentiment classification task. Therefore, to enhance SA classifiers' performance, we propose a novel semi-supervised Sentiment Hashtag Embedding (SHE) model in the second contribution (Chapter 5) to encode sentiment information while preserving the semantic characteristics of hashtags. In particular, we exploit multitask learning approach through Autoencoder and Convolutional Neural Network classifier to train the proposed SHE model. From various experimental evaluations, it is observed that SHE yields robust hashtag embeddings and performs better than state-of-the-art baselines. It is also observed that SHE can be effectively used for various tasks like hashtag sentiment classification, tweet sentiment classification, hashtag retrieval, and sentiment hashtag lexicon generation for non-English languages.

Sentiment classification on tweets often needs to deal with the problems of under-specificity, noise, and multilingual content. To address the above challenges, the third contribution of the thesis (Chapter 6) proposes a heterogeneous multi-layer network-based representation of tweets to generate multiple representations of a tweet. Further, we propose a centrality aware random-walk for node embedding and tweet representations suitable for the multi-layer network. Our proposed centrality aware random-walk method can generate walk sequences that capture better semantic relations than its unbiased and biased random walk based counterparts. From various experimental observations, it is evident that sentiment-oriented node expansion can reduce under-specificity, noise in a tweet, and enhance the representation. The proposed method outperforms its text-based counterpart in a majority of the cases.

Though the performance of the sentiment classification improves with the

above contributions, a systematic study of incorporating textual and structural features using various state-of-the-art embedding methods have not been investigated. The final contribution of the thesis (Chapter 7) proposes an approach to incorporate the text-based and graph-based representations of tweet through *end-to-end* learning framework for sentiment classification task. From various experimental analyses, it is evident that incorporating both text-based and graph-based representations can address under-specificity, noisy text, and multilingual content and provide better classification performance than its counterparts.

8.2 Future scope of research

The work presented in the chapters of this thesis contributes broad scope and proclaims several directions for future research endeavors. This section discusses some of the potential directions for future extension of the thesis work.

Hashtag based sentiment analysis: Sentiment analysis of tweets require a huge amount of annotated corpus for a particular domain. Besides, curating sentiment annotated resources is an expensive task. As hashtags are user annotated meta-information of the tweets, utilizing sentiment hashtags can gather huge amount of sentiment annotated resources. In future, efforts may be directed towards identifying and utilizing sentiment hashtags for building sentiment classifier. The second contribution (Chapter 5) of the thesis work i.e., Sentiment Hashtag Embedding (SHE) can be exploited for identifying and utilizing sentiment hashtags and evaluate the efficacy of the sentiment classifier performance.

Multilingual sentiment analysis: Although most of the publicly available sentiment lexicons are in the English language, people tend to express sentiment

polarized opinions using their own local language[2]. For example, a multilingual positive sentiment tweet *"@narendramodi Thank you Sir GST laagu karne ke liye is India great"* – is a code-mixed of Hindi and English languages where the author of the tweet is praising another twitterer *@narendramodi* for implementing GST*. It is observed from Chapter 6 and 7 that our proposed models can address multilingual content of tweets through network expansion of tweets. This works can be extended to explore the problem of sentiment analysis of multilingual tweets.

Identifying the role of entities in sentiment tweets: It is observed from the first contribution (Chapter 4) of the thesis work that the sentiment classifiers have low performance due to the presence of different natures of tweet such as stance, aspect-based, sarcastic, and code-mixed language tweets in the Societal dataset. Identifying the role of entities involved through sentiment analysis of tweets can help in identifying the stance of the user opinions as well as sarcastic tweets.

*Goods and Services Tax

References

[1] Abbasi, A., Hassan, A., & Dhar, M. (2014). Benchmarking twitter sentiment analysis tools. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*, volume 14 (pp. 26–31).

[2] Agarwal, P., Sharma, A., Grover, J., Sikka, M., Rudra, K., & Choudhury, M. (2017). I may talk in english but gaali toh hindi mein hi denge: A study of english-hindi code-switching and swearing pattern on social networks. In *2017 9th International Conference on Communication Systems and Networks (COMSNETS)* (pp. 554–557).: IEEE.

[3] Akhtar, M. S., Kumar, A., Ekbal, A., & Bhattacharyya, P. (2016). A hybrid deep learning architecture for sentiment analysis. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING): Technical Papers* (pp. 482–493).

[4] Al-Smadi, M., Qawasmeh, O., Al-Ayyoub, M., Jararweh, Y., & Gupta, B. (2017). Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of arabic hotels' reviews. *Journal of Computational Science*.

[5] Al-Twairesh, N. & Al-Negheimish, H. (2019). Surface and deep features ensemble for sentiment analysis of arabic tweets. *IEEE Access*, 7, 84122–84131.

[6] Alfina, I., Sigmawaty, D., Nurhidayati, F., & Hidayanto, A. N. (2017). Utilizing hashtags for sentiment analysis of tweets in the political domain. In *Proceedings of the 9th International Conference on Machine Learning and Computing* (pp. 43–47).: ACM.

[7] Alsaedi, N. & Burnap, P. (2015). Feature extraction and analysis for identifying disruptive events from social media. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015* (pp. 1495–1502).: ACM.

[8] Araque, O., Corcuera-Platas, I., Sánchez-Rada, J. F., & Iglesias, C. A. (2017). Enhancing deep learning sentiment analysis with ensemble techniques in social applications. *Expert Systems with Applications*, 77, 236–246.

[9] Balamurali, A., Joshi, A., & Bhattacharyya, P. (2011). Harnessing wordnet senses for supervised sentiment classification. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing* (pp. 1081–1091).: Association for Computational Linguistics.

[10] Barbosa, L. & Feng, J. (2010). Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters* (pp. 36–44).: Association for Computational Linguistics.

[11] Bijari, K., Zare, H., Kebriaei, E., & Veisi, H. (2020). Leveraging deep graph-based text representation for sentiment polarity applications. *Expert Systems with Applications*, 144, 113090.

[12] Binali, H., Potdar, V., & Wu, C. (2009). A state of the art opinion mining and its application domains. In *2009 IEEE International Conference on Industrial Technology* (pp. 1–6).: IEEE.

[13] Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5, 135–146.

[14] Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual web search engine. *Computer Networks*, 30, 107–117.

[15] Burnap, P., Williams, M. L., Sloan, L., Rana, O., Housley, W., Edwards, A., Knight, V., Procter, R., & Voss, A. (2014). Tweeting the terror: modelling the social media reaction to the woolwich terrorist attack. *Social Network Analysis and Mining*, 4(1), 206.

[16] Cao, X., MacNaughton, P., Deng, Z., Yin, J., Zhang, X., & Allen, J. G. (2018). Using twitter to better understand the spatiotemporal patterns of public sentiment: A case study in massachusetts, usa. *International journal of environmental research and public health*, 15(2), 250.

[17] Carbonell, J. G. (1979). *Subjective Understanding: Computer Models of Belief Systems.* Technical report, YALE UNIV NEW HAVEN CONN DEPT OF COMPUTER SCIENCE.

[18] Caruana, R. (1997). Multitask learning. *Machine learning*, 28(1), 41–75.

[19] Catal, C. & Nangir, M. (2017). A sentiment classification model based on multiple classifiers. *Applied Soft Computing*, 50, 135–141.

[20] Cen, Y., Zou, X., Zhang, J., Yang, H., Zhou, J., & Tang, J. (2019). Representation learning for attributed multiplex heterogeneous network. In *Proceedings of the 25th ACM International Conference on Knowledge Discovery & Data Mining* (pp. 1358–1368).

[21] Chen, J., Hou, H., Ji, Y., & Gao, J. (2019). Graph convolutional networks with structural attention model for aspect based sentiment analysis. In *2019 International Joint Conference on Neural Networks (IJCNN)* (pp. 1–7).: IEEE.

[22] Chen, S., Mao, J., Li, G., Ma, C., & Cao, Y. (2020). Uncovering sentiment and retweet patterns of disaster-related tweets from a spatiotemporal perspective–a case study of hurricane harvey. *Telematics and Informatics*, 47, 101326.

[23] Chen, T., Xu, R., He, Y., & Wang, X. (2017). Improving sentiment analysis via sentence type classification using bilstm-crf and cnn. *Expert Systems with Applications*, 72, 221–230.

[24] Church, K. W. & Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1), 22–29.

[25] Collobert, R. & Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning* (pp. 160–167).: ACM.

[26] Cui, H., Mittal, V., & Datar, M. (2006). Comparative experiments on sentiment classification for online product reviews. In *Proceedings of the 21st National Conference on Artificial Intelligence, AAAI Press*, volume 2 (pp. 1265–1270).

[27] Dahou, A., Xiong, S., Zhou, J., Haddoud, M. H., & Duan, P. (2016). Word embeddings and convolutional neural network for arabic sentiment classification. In *Proceedings of coling 2016, the 26th international conference on computational linguistics: Technical papers* (pp. 2418–2427).

[28] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

[29] dos Santos, C. & Gatti, M. (2014). Deep convolutional neural networks for sentiment analysis of short texts. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING): Technical Papers* (pp. 69–78).

[30] Fu, P., Lin, Z., Yuan, F., Wang, W., & Meng, D. (2018a). Learning sentiment-specific word embedding via global sentiment representation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018* (pp. 4808–4815).

[31] Fu, P., Lin, Z., Yuan, F., Wang, W., & Meng, D. (2018b). Learning sentiment-specific word embedding via global sentiment representation. In *Thirty-Second AAAI Conference on Artificial Intelligence* (pp. 4808–4815).

[32] Garay, J., Yap, R., & Sabellano, M. (2019). An analysis on the insights of the anti-vaccine movement from social media posts using k-means clustering algorithm and vader sentiment analyzer. *IOP Conference Series: Materials Science and Engineering*, 482(1), 012043.

[33] Garg, P., Garg, H., & Ranga, V. (2017). Sentiment analysis of the uri terror attack using twitter. In *Computing, Communication and Automation (ICCCA), 2017 International Conference on* (pp. 17–20).: IEEE.

[34] Ghorbani, M., Baghshah, M. S., & Rabiee, H. R. (2019). Mgcn: semi-supervised classification in multi-layer graphs with graph convolutional networks. In *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (pp. 208–211).

[35] Giachanou, A. & Crestani, F. (2016). Like it or not: A survey of twitter sentiment analysis methods. *ACM Computing Surveys (CSUR)*, 49(2), 28.

[36] Go, A., Bhayani, R., & Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12), 2009.

[37] Goldberg, Y. (2016). A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 57, 345–420.

[38] Gonçalves, P., Araújo, M., Benevenuto, F., & Cha, M. (2013). Comparing and combining sentiment analysis methods. In *Proceedings of the first ACM conference on Online social networks* (pp. 27–38).

[39] Goyal, A. & Daumé III, H. (2011). Generating semantic orientation lexicon using large data and thesaurus. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis* (pp. 37–43).: Association for Computational Linguistics.

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

[41] Gui, L., Zhou, Y., Xu, R., He, Y., & Lu, Q. (2017). Learning representations from heterogeneous network for sentiment classification of product reviews. *Knowledge-Based Systems*, 124, 34–45.

[42] Hanteer, O. & Rossi, L. (2019). An innovative way to model twitter topic-driven interactions using multiplex networks. *Frontiers in Big Data*, 2, 9.

[43] Hassan, A. & Radev, D. (2010). Identifying text polarity using random walks. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 395–403).: Association for Computational Linguistics.

[44] Jianqiang, Z., Xiaolin, G., & Xuejun, Z. (2018). Deep convolution neural networks for twitter sentiment analysis. *IEEE Access*, 6, 23253–23260.

[45] Joty, S. R., Màrquez, L., & Nakov, P. (2018). Joint multitask learning for community question answering using task-specific embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018* (pp. 4196–4207).

[46] Karamibekr, M. & Ghorbani, A. A. (2012). Sentiment analysis of social issues. In *Proceedings of the International Conference on Social Informatics (SocialInformatics)* (pp. 215–221).

[47] Kilgarriff, A. & Rose, T. (1998). Measures for corpus similarity and homogeneity. In *Proceedings of the Third Conference on Empirical Methods for Natural Language Processing* (pp. 46–52).

[48] Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1746–1751).: Association for Computational Linguistics.

[49] Kingma, D. P. & Ba, J. (2015). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings*.

[50] Kipf, T. N. & Welling, M. (2017). Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*: OpenReview.net.

[51] Kiritchenko, S., Zhu, X., Cherry, C., & Mohammad, S. (2014). Nrc-canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)* (pp. 437–442).

[52] Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J. P., Moreno, Y., & Porter, M. A. (2014). Multilayer networks. *Journal of complex networks*, 2(3), 203–271.

[53] Kouloumpis, E., Wilson, T., & Moore, J. D. (2011). Twitter sentiment analysis: The good the bad and the omg! In *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)* (pp. 164).

[54] Krokos, E., Samet, H., & Sankaranarayanan, J. (2014). A look into twitter hashtag discovery and generation. In *Proceedings of the 7th ACM SIGSPATIAL International Workshop on Location-Based Social Networks* (pp. 49–56).: ACM.

[55] Kušen, E. & Strembeck, M. (2018). Politics, sentiments, and misinformation: An analysis of the twitter discussion on the 2016 austrian presidential elections. *Online Social Networks and Media*, 5, 37–50.

[56] Lafferty, J. & Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 111–119).

[57] Lee, J. Y. & Dernoncourt, F. (2016). Sequential short-text classification with recurrent and convolutional neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 515–520).: Association for Computational Linguistics.

[58] Leman, A. & Weisfeiler, B. (1968). A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsiya*, 2(9), 12–16.

[59] Lerman, K., Arora, M., Gallegos, L., Kumaraguru, P., & Garcia, D. (2016). Emotions, demographics and sociability in twitter interactions. In *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)* (pp. 201–210).

[60] Li, J., Chen, C., Tong, H., & Liu, H. (2018). Multi-layered network embedding. In *Proceedings of the 2018 SIAM International Conference on Data Mining* (pp. 684–692).: SIAM.

[61] Li, Y. & Patra, J. C. (2010). Genome-wide inferring gene–phenotype relationship by walking on the heterogeneous network. *Bioinformatics*, 26(9), 1219–1224.

[62] Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1), 1–167.

[63] Liu, B. (2015). *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press.

[64] Liu, J., He, Z., & Huang, Y. (2018). Hashtag2vec: Learning hashtag representation with relational hierarchical embedding model. In *IJCAI* (pp. 3456–3462).

[65] Liu, J. & Zhang, Y. (2017). 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* (pp. 572–577).

[66] Long, M., Cao, Z., Wang, J., & Philip, S. Y. (2017). Learning multiple tasks with multilinear relationship networks. In *Advances in neural information processing systems* (pp. 1594–1603).

[67] Lu, J., Behbood, V., Hao, P., Zuo, H., Xue, S., & Zhang, G. (2015). Transfer learning using computational intelligence: A survey. *Knowledge-Based Systems*, 80, 14–23.

[68] Lu, Y., Sakamoto, K., Shibuki, H., & Mori, T. (2017). Are deep learning methods better for twitter sentiment analysis? In *Proceedings of the 23rd Annual Meeting of Natural Language Processing (Japan)* (pp. 787–790).

[69] Lu, Z., Du, P., & Nie, J.-Y. (2020). Vgcn-bert: augmenting bert with graph embedding for text classification. In *European Conference on Information Retrieval* (pp. 369–382).: Springer.

[70] Luo, D., Ni, J., Wang, S., Bian, Y., Yu, X., & Zhang, X. (2020). Deep multi-graph clustering via attentive cross-graph association. In *Proceedings of the 13th International Conference on Web Search and Data Mining* (pp. 393–401).

[71] Ma, Z., Sun, A., Yuan, Q., & Cong, G. (2014). Tagging your tweets: A probabilistic modeling of hashtag annotation in twitter. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management* (pp. 999–1008).: ACM.

[72] Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, volume 1 (pp. 142–150).

[73] Maynard, D. & Bontcheva, K. (2016). Challenges of evaluating sentiment analysis tools on social media. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)* (pp. 1142–1148).: LREC.

[74] McAuley, J. & Leskovec, J. (2013). Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems* (pp. 165–172).

[75] Meng, F., Feng, J., Yin, D., Chen, S., & Hu, M. (2020). Sentiment analysis with weighted graph convolutional networks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings* (pp. 586–595).

[76] Meng, J. T., Shang, J., Ren, X., Zhang, M., & Han, J. (2017). An attention-based collaboration framework for multi-view network representation learning. *CIKM. ACM.*

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

[78] Mohammad, S., Dunne, C., & Dorr, B. (2009a). Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus.

In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2* (pp. 599–608).: Association for Computational Linguistics.

[79] Mohammad, S., Dunne, C., & Dorr, B. (2009b). Generating high-coverage semantic orientation lexicons from overtly marked words and a thesaurus. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, EMNLP '09 (pp. 599–608). Stroudsburg, PA, USA: Association for Computational Linguistics.

[80] Mohammad, S. M. & Kiritchenko, S. (2015). Using hashtags to capture fine emotion categories from tweets. *Computational Intelligence*, 31(2), 301–326.

[81] Mohammad, S. M., Sobhani, P., & Kiritchenko, S. (2017). Stance and sentiment in tweets. *ACM Transactions on Internet Technology (TOIT)*, 17(3), 26.

[82] Mohammad, S. M. & Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, 29(3), 436–465.

[83] Mohammad, S. M., Zhu, X., Kiritchenko, S., & Martin, J. (2015). Sentiment, emotion, purpose, and style in electoral tweets. *Information Processing & Management*, 51(4), 480–499.

[84] Mostafa, A. M. (2017). An evaluation of sentiment analysis and classification algorithms for arabic textual data. *International Journal of Computer Applications*, 158(3).

[85] Muhammad, A., Wiratunga, N., & Lothian, R. (2016). Contextual sentiment analysis for social media genres. *Knowledge-based systems*, 108, 92–101.

[86] Nayak, A., Timmapathini, H., Ponnalagu, K., & Venkoparao, V. G. (2020). Domain adaptation challenges of bert in tokenization and sub-word representations of out-of-vocabulary words. In *Proceedings of the First Workshop on Insights from Negative Results in NLP* (pp. 1–5).

[87] Neppalli, V. K., Caragea, C., Squicciarini, A., Tapia, A., & Stehle, S. (2017). Sentiment analysis during hurricane sandy in emergency response. *International Journal of Disaster Risk Reduction*, 21, 213–222.

[88] Nguyen, H. & Nguyen, M.-L. (2017). A deep neural architecture for sentence-level sentiment classification in twitter social networking. In *International Conference of the Pacific Association for Computational Linguistics* (pp. 15–27).: Springer.

[89] Ni, J., Chang, S., Liu, X., Cheng, W., Chen, H., Xu, D., & Zhang, X. (2018). Co-regularized deep multi-network embedding. In *Proceedings of the 2018 World Wide Web Conference* (pp. 469–478).

[90] On, J., Park, H.-A., & Song, T.-M. (2019). Sentiment analysis of social media on childhood vaccination: Development of an ontology. *Journal of medical Internet research*, 21(6), e13456.

[91] Ouyang, X., Zhou, P., Li, C. H., & Liu, L. (2015). Sentiment analysis using convolutional neural network. In *Proceedings of the IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM)* (pp. 2359–2364).

[92] Öztürk, N. & Ayvaz, S. (2018). Sentiment analysis on twitter: A text mining approach to the syrian refugee crisis. *Telematics and Informatics*, 35(1), 136–147.

[93] Pak, A. & Paroubek, P. (2010). Twitter as a corpus for sentiment analysis and opinion mining. In *LREc*, volume 10 (pp. 1320–1326).

[94] Pang, B., Lee, L., et al. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval*, 2(1–2), 1–135.

[95] Pang, B., Lee, L., & Vaithyanathan, S. (2002). Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing (EMNLP)*, volume 10 (pp. 79–86).

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

[97] Pham, D.-H. & Le, A.-C. (2018). Learning multiple layers of knowledge representation for aspect based sentiment analysis. *Data & Knowledge Engineering*, 114, 26–39.

[98] Poria, S., Cambria, E., & Gelbukh, A. (2016). Aspect extraction for opinion mining with a deep convolutional neural network. *Knowledge-Based Systems*, 108, 42–49.

[99] Prabowo, R. & Thelwall, M. (2009). Sentiment analysis: A combined approach. *Journal of Informetrics*, 3(2), 143–157.

[100] Qadir, A. & Riloff, E. (2013). Bootstrapped learning of emotion hashtags# hashtags4you. In *Proceedings of the 4th workshop on computational approaches to subjectivity, sentiment and social media analysis* (pp. 2–11).

[101] Qadir, A. & Riloff, E. (2014). Learning emotion indicators from tweets: Hashtags, hashtag patterns, and phrases. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1203–1209).

[102] Qu, M., Tang, J., Shang, J., Ren, X., Zhang, M., & Han, J. (2017). An attention-based collaboration framework for multi-view network representation learning. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (pp. 1767–1776).

[103] Rahmede, C., Iacovacci, J., Arenas, A., & Bianconi, G. (2018). Centralities of nodes and influences of layers in large multiplex networks. *Journal of Complex Networks*, 6(5), 733–752.

[104] Ribeiro, F. N., Araújo, M., Gonçalves, P., Gonçalves, M. A., & Benevenuto, F. (2016). Sentibench-a benchmark comparison of state-of-the-practice sentiment analysis methods. *EPJ Data Science*, 5(1), 23.

[105] Ruder, S. (2017). An overview of multi-task learning in deep neural networks. *CoRR*, abs/1706.05098.

[106] Saif, H., He, Y., Fernandez, M., & Alani, H. (2014). Semantic patterns for sentiment analysis of twitter. In *International Semantic Web Conference* (pp. 324–340).: Springer.

[107] Saif, H., He, Y., Fernandez, M., & Alani, H. (2016). Contextual semantics for sentiment analysis of twitter. *Information Processing & Management*, 52(1), 5–19.

[108] Schütze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.

[109] Severyn, A. & Moschitti, A. (2015). Twitter sentiment analysis with deep convolutional neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 959–962).

[110] Shannon, C. E. (2001). A mathematical theory of communication. *ACM SIGMOBILE Mobile Computing and Communications Review*, 5(1), 3–55.

[111] Shi, H.-X. & Li, X.-J. (2011). A sentiment analysis model for hotel reviews based on supervised learning. In *2011 International Conference on Machine Learning and Cybernetics*, volume 3 (pp. 950–954).: IEEE.

[112] Silva, N. F. F. D., Coletta, L. F., & Hruschka, E. R. (2016). A survey and comparative study of tweet sentiment analysis via semi-supervised learning. *ACM Computing Surveys (CSUR)*, 49(1), 1–26.

[113] Singh, L. G., Anil, A., & Singh, S. R. (2020). She: Sentiment hashtag embedding through multitask learning. *IEEE Transactions on Computational Social Systems*, 7(2), 417–424.

[114] Singh, L. G., Mitra, A., & Singh, S. R. (2020). Sentiment analysis of tweets using heterogeneous multi-layer network representation and embedding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 8932–8946).

[115] Singh, L. G. & Singh, S. R. (2020). Empirical study of sentiment analysis tools and techniques on societal topics. *Journal of Intelligent Information Systems*, (pp. 1–29).

[116] Singh, P., Sawhney, R. S., & Kahlon, K. S. (2018). Sentiment analysis of demonetization of 500 & 1000 rupee banknotes by indian government. *ICT Express*, 4(3), 124–129.

[117] Sobhani, P., Mohammad, S., & Kiritchenko, S. (2016). Detecting stance in tweets and analyzing its interaction with sentiment. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics* (pp. 159–169).

[118] Taboada, M., Brooke, J., Tofiloski, M., Voll, K., & Stede, M. (2011). Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2), 267–307.

[119] Tang, D., Wei, F., Qin, B., Yang, N., Liu, T., & Zhou, M. (2016). Sentiment embeddings with applications to sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering*, 28(2), 496–509.

[120] Tang, D., Wei, F., Yang, N., Zhou, M., Liu, T., & Qin, B. (2014). Learning sentiment-specific word embedding for twitter sentiment classification. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1 (pp. 1555–1565).

[121] Tsitsulin, A., Mottin, D., Karras, P., & Müller, E. (2018). Verse: Versatile graph embeddings from similarity measures. In *Proceedings of the 2018 World Wide Web Conference on World Wide Web* (pp. 539–548).

[122] Tumasjan, A., Sprenger, T. O., Sandner, P. G., & Welpe, I. M. (2010). Predicting elections with twitter: What 140 characters reveal about political sentiment. *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)*, 10(1), 178–185.

[123] Turney, P. D. (2002). Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th annual meeting on Association for Computational Linguistics* (pp. 417–424).: Association for Computational Linguistics.

[124] Turney, P. D. & Littman, M. L. (2003). Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, 21(4), 315–346.

[125] Vania, C., Ibrahim, M., & Adriani, M. (2014). Sentiment lexicon generation for an under-resourced language. *Int. J. Comput. Linguistics Appl.*, 5(1), 59–72.

[126] Vargas, S., McCreadie, R., Macdonald, C., & Ounis, I. (2016). Comparing overall and targeted sentiments in social media during crises. In *Proceedings of the International AAAI Conference on Web and Social Media (ICWSM)* (pp. 695–698).

[127] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998–6008).

[128] Vilares, D., Alonso, M. A., & Gómez-Rodríguez, C. (2017). Supervised sentiment analysis in multilingual environments. *Information Processing & Management*, 53(3), 595–607.

[129] Violos, J., Tserpes, K., Psomakelis, E., Psychas, K., & Varvarigou, T. (2016). Sentiment analysis using word-graphs. In *Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics* (pp. 1–9).

[130] Wang, G., Sun, J., Ma, J., Xu, K., & Gu, J. (2014). Sentiment classification: The contribution of ensemble learning. *Decision support systems*, 57, 77–93.

[131] Wang, X., Wei, F., Liu, X., Zhou, M., & Zhang, M. (2011). Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach.

In *Proceedings of the 20th ACM international conference on Information and knowledge management* (pp. 1031–1040).: ACM.

[132] Wang, Y., Liu, J., Huang, Y., & Feng, X. (2016). Using hashtag graph-based topic model to connect semantically-related words without co-occurrence in microblogs. *IEEE Transactions on Knowledge and Data Engineering*, 28(7), 1919–1933.

[133] Weston, J., Chopra, S., & Adams, K. (2014). # tagspace: Semantic embeddings from hashtags. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1822–1827).

[134] Wilks, Y. & Bien, J. (1983). Beliefs, points of view, and multiple environments. *Cognitive Science*, 7(2), 95–119.

[135] Xia, R., Zong, C., & Li, S. (2011). Ensemble of feature sets and classification algorithms for sentiment classification. *Information Sciences*, 181(6), 1138–1152.

[136] Xu, G., Meng, Y., Qiu, X., Yu, Z., & Wu, X. (2019). Sentiment analysis of comment texts based on bilstm. *IEEE Access*, 7, 51522–51532.

[137] Yao, L., Mao, C., & Luo, Y. (2019). Graph convolutional networks for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33 (pp. 7370–7377).

[138] Yazidi, A., Bai, A., Hammer, H., & Engelstad, P. (2015). A simple and efficient algorithm for lexicon generation inspired by structural balance theory. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems* (pp. 336–347).: Springer.

[139] Ye, Z., Li, F., & Baldwin, T. (2018). Encoding sentiment information into word vectors for sentiment analysis. In *Proceedings of the 27th International Conference on Computational Linguistics* (pp. 997–1007).

[140] Zarrella, G. & Marsh, A. (2016). Mitre at semeval-2016 task 6: Transfer learning for stance detection. *Proceedings of SemEval*, (pp. 458–463).

[141] Zhang, C., Li, Q., & Song, D. (2019). Aspect-based sentiment classification with aspect-specific graph convolutional networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 4568–4578).

[142] Zhang, H., Qiu, L., Yi, L., & Song, Y. (2018a). Scalable multiplex network embedding. In *IJCAI*, volume 18 (pp. 3082–3088).

[143] Zhang, J. (2020). Segmented graph-bert for graph instance modeling. *arXiv preprint arXiv:2002.03283*.

[144] Zhang, L., Wang, S., & Liu, B. (2018b). Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, (pp. e1253).

[145] Zhang, M., Cui, Z., Neumann, M., & Chen, Y. (2018c). An end-to-end deep learning architecture for graph classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

[146] Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28, 649–657.

[147] Zhang, Y., Lai, G., Zhang, M., Zhang, Y., Liu, Y., & Ma, S. (2014). Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval* (pp. 83–92).

[148] Zhao, Z., Lu, H., Cai, D., He, X., & Zhuang, Y. (2017). Microblog sentiment classification via recurrent random walk network learning. In *IJCAI*, volume 17 (pp. 3532–3538).

[149] Zhou, G.-Y. & Huang, J. X. (2017). Modeling and mining domain shared knowledge for sentiment analysis. *ACM Transactions on Information Systems (TOIS)*, 36(2), 1–36.

[150] Zhu, X. & Ghahramani, Z. (2002). Learning from labeled and unlabeled data with label propagation.

[151] Zitnik, M. & Leskovec, J. (2017). Predicting multicellular function through multi-layer tissue networks. *Bioinformatics*, 33(14), i190–i198.

Publications

Journals

1. **Loitongbam Gyanendro Singh**, Akash Anil, and Sanasam Ranbir Singh. *SHE: Sentiment Hashtag Embedding Through Multitask learning* In *IEEE Transaction on Computational Social System* Volume: 7, Issue: 2, April 2020, pages 417-424

2. **Loitongbam Gyanendro Singh** and Sanasam Ranbir Singh. *Empirical Study of Sentiment Analysis Tools and Techniques on Societal Topics.* In *Journal of Intelligent Information Systems* October 2020, pages 1-29

3. **Loitongbam Gyanendro Singh** and Sanasam Ranbir Singh. *Characteristics of Societal and non-Societal dataset.* [Preparing for submission]

Conferences

1. **Loitongbam Gyanendro Singh**, Anasua Mitra, and Sanasam Ranbir Singh. *Sentiment Analysis of Tweets using Heterogeneous Multi-layer Network Representation and Embedding.* In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Association for Computational Linguistics 2020 pages 8932–8946.

2. **Loitongbam Gyanendro Singh** and Sanasam Ranbir Singh. *Sentiment Analysis Tweets using Multiview learning approach* [Under review]

Other publications (Not related to thesis work)

1. **Loitongbam Gyanendro Singh**, Lenin Laitonjam, and Sanasam Ranbir Singh. *Automatic Syllabification for Manipuri Language.* In *Proceedings of the COLING 2016, the 26th International Conference on Computational Linguistics*,

Association for Computational Linguistics 2016 pages 349-357.

2. **Loitongbam Gyanendro Singh**, Nagaraj Adiga, Bidisha Sharma, Sanasam Ranbir Singh, and S R M Prasanna. *Automatic Pause Marking for Speech Synthesis.* In *Proceedings of the TENCON 2017-2017 IEEE Region 10 Conference*, IEEE 2017 pages 1790-1794.

3. **Loitongbam Gyanendro Singh** and Sanasam Ranbir Singh. *Word Polarity Detection using Syllable Features for Manipuri Language.* In *Proceedings of 2017 International Conference on Asian Language Processing (IALP)*, IEEE 2017 pages 206–209.

4. Lenin Laitonjam, **Loitongbam Gyanendro Singh**, and Sanasam Ranbir Singh. *Transliteration of English Loanwords and Named-entities to Manipuri: Phoneme vs Grapheme representation.* In *Proceedings of the 2018 International Conference on Asian Language Processing (IALP)*, IEEE 2018 pages 255-260.

5. Durgesh Kumar, **Loitongbam Gyanendro Singh**, and Sanasam Ranbir Singh. *Hashtag Based Tweet Expansion for Improved Topic Modeling.* In *IEEE Transaction on Computational Social System* [Under review]

6. Anasua Mitra, **Loitongbam Gyanendro Singh**, Roshan Singh, Durgesh Kumar, Sanasam Ranbir Singh, and Sathish Kumar. *Correlating Social Network Activity and Physical Network Activity through Mobility Patterns and Sentiments* [Under review]