INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

Significance of Hashtags for Improved Topic Modeling on Tweets

by

Durgesh Kumar

A thesis submitted in partial fulfillment for the degree of Doctor of Philosophy

in the

Department of Computer Science and Engineering

Under the supervision of

Dr. Sanasam Ranbir Singh

March 2022

Declaration of Authorship

I, Durgesh Kumar, hereby confirm that:

- The work contained in this thesis is original and has been done by myself under the general supervision of my supervisor.
- This work has not been submitted to any other Institute for any degree or diploma.
- Whenever I have used materials (data, theoretical analysis, results) from other sources, I have given due credit to the authors/researchers by citing them in the text of the thesis and giving their details in the reference.
- Whenever I have quoted from the work of others, the source is always given.

Durgesh Kumar
Research Scholar,
Department of CSE,
Indian Institute of Technology Guwahati,
Guwahati, Assam, INDIA 781039,
durgeshit@gmail.com, k.durgesh@iitg.ac.in

Place: IIT Guwahati

Certificate

This is to certify that the thesis entitled **"Significance of Hashtags for Improved Topic Modeling on Tweets"** being submitted by **Mr. Durgesh Kumar** to the department of *Computer science and Engineering, Indian Institute of Technology Guwahati*, is a record of bonafide research work under my supervision and is worthy of consideration for the award of the degree of Doctor of Philosophy of the Institute.

Dr. Sanasam Ranbir Singh
Department of CSE,
Indian Institute of Technology Guwahati,
Guwahati, Assam, INDIA 781039,
ranbir@iitg.ac.in

Place: IIT Guwahati

Dedicated to

my father Late Binod Bhagat

and

my mother Smt. Gita Devi

& all of my teachers and gurus

for their infinite love, support, motivation and guidance.

Acknowledgements

It gives me immense pleasure to thank each individual who supported directly or indirectly towards completion of my Ph.D. journey. At first, I would like to thank my supervisor Dr. Sanasam Ranbir Singh for his exceptional and motivating guidance throughout my Ph.D. journey. Moreover, his dedication towards his duties and research always inspire me. I would always be indebted to him for several thought provoking ideas, constructive research discussions and cultivating professional work ethics.

I would like to thank my Doctoral Committee members namely, Prof. Sukumar Nandi, Dr. Ashish Anand, Dr. V. Vijaya Saradhi, and Dr. Prithwijit Guha for their constructive suggestions towards shaping my research goals as well as the entire thesis. I also want to thank Dr. Ashish Anand for the various research discussion and his invaluable suggestions. Furthermore, my sincere thanks to Prof. Jatindra Kumar Deka, the Head of the Department of Computer Science and Engineering and other faculty members for their direct and indirect support.

I humbly thank to Mr. Raktajit Pathak, Mr. Nanu Alan Kachari, Mr. Bhriguraj Borah and all other institute's staffs for all the helps I borrowed towards making my journey smooth and productive. I specially thank Mr. Nanu Alan Kachari and Mr. Bhriguraj Borah for their extreme dedications towards managing efficient computing facilities at the department. My thesis would not have been completed without their timely support. Furthermore, I would like to thank IIT Guwahati administration for providing on-campus hostel facility. From the core of my heart, I would like to thank the the administrative staffs from Students Affairs section, Academic Section, Research and Development section, Hostel caretakers, mess staffs, canteen staffs, security personals, and housekeeping staffs for making my stay memorable and smooth.

Having good friends is always a blessing. Fortunately, I have a large set of good and close friends with whom I have spent very quality time. I am privileged to mention Sai Manoj Yadlapati, Dr. Akash Anil, Dr. Khushboo Rani, Abha Kumari as four longtime peers who supported my entire journey in several perspectives. I had the privilege of having a very helpful and supportive seniors (as friends), namely Dr. Niladri Sett, Dr. Sounak Chakraborty, Dr. Satish Kumar, Dr. Jitendra Kumar, Dr. Sibaji Gaj, Kunwer Mrityunjay Singh, Dr. Awnish Kumar, and Mausam Handique. I would like to mention Dr. Niladri Sett and Dr. Akash Anil, Dr. Khushboo Rani, Abha Kumari especially for their critical comments

and road-maps which made my Ph.D. journey smoother and efficient. I really feel privileged to have many joyous moments with Satish, Avinash, Sumit, Jitu, Saloni, Deepika, Nayan, Tushar, Sarthak, Nisha, Shivam, Amit, Vikrant, Harsha, Ila, Arvin, Ruchika, Shruti, Garima, Gourangi, Khushboo, Gajendra, Divesh and Rishi. My stay at IIT Guwahati was made more pleasant by having many good memories with friends from OSINT lab like Akash, Neelakshi, Hemanta, Gyanendro, Sujit, Lenin, Bornali, Mala, Pankaj, Tonmoya, Deepen, Anupam, Jubanjan, Anasua, Rajib Sir, Piyush, Pardeep, Neelesh, Nitesh, Ranjan, Rakesh, Rajlakshmi, Akhilesh, Debashish and many more. Further, I am thankful to friends and juniors at IIT Guwahati such as Amit Raj, Amit Bhagel, Subhash Pratap Singh, Vinod Vishwakarma, Amit Khoiwal, Vikrant Singh, Umesh Chowdhary, Ravi Kumar.

During my Ph.D. Journey, I was fortunate to work with many creative minds of IIT Guwahati. Some of them are Sai Manoj, Dr. Akash Anil, Dr. Khushboo Rani, Dr. Neelaskhi Sharma, Dr. Gyannedro, Dr. Debashish Naskar, Jyotindra Narayan, Rahul Raoniar, Sujit Kumar, Anasua Mitra, Ranjan Sharma, Nitesh Bhattacharya, and Rakesh Singha. Our countless discussions and dedications paved a fruitful way to shape my Ph.D. Furthermore, I would like to thank all the anonymous reviewers of my papers & thesis as well as friends I forgot to mention here.

I would like to thank all the doctors, Kerala Ayurveda and Art of Living family for taking care of my overall health. I specially want to thank Dr. Anuj Kumar Barua, Dr. Chintu Barman, and Dr. Lakshmi Chaya for helping me in regaining my health after severe knee injury and face paralysis. I also want to convey my deep gratitude to Mr. Deepen Mukherjee, Dr. Virat Chirania, Mr. Vishnu Praskash, Mr. Vivek Agarawal, Mr. Suvidha Agarawal, Mr. Avinash Tiku, Dr. Atreyi Ghosh, Miss Puja Sharma for guiding me in the spiritual life and mentoring me to tackle all the challenges of life gracefully.

My family members have played a crucial role throughout my academic carrier. Their constant support, love, motivation, and faith in me help me bounce back at different phases of my life. From the core of my heart, I would like to thank my parents, my sister Miss Ratna Kumari, my brother-in-law Mr. Vicky Bhagat, my niece Arag Arpan, Nisha, and Komal for showering immense love and moral support. Finally, I would like to thank all the friends, family, friends, relatives, and everyone who has supported my academic carrier.

Abstract

With the increase in Twitter's popularity, topic modeling on Twitter has become an important problem with applications in diverse fields such as text summarization, document clustering, information retrieval, and sentiment analysis. The short and noisy tweets with informal writing style make topic modeling on tweets more challenging due to increased data sparsity and under-specificity. Latent Dirichlet Allocation (LDA), one of the widely used topic models, suffers from data sparsity and under-specificity. Researchers have tried to counter the data sparsity and under-specificity in tweets by adding related content from external sources such as News pages and Wikipedia or pooling related tweets to pseudo documents. Adding the content from external resources is non-trivial due to differences in writing styles and vocabulary. Moreover, Topic modeling on pooled documents may lose the distribution of topics over the individual tweet and increase the corpus size due to duplicate tweets in different pools. From earlier studies and our preliminary investigation, hashtags are found to provide necessary meta-information in linking tweets to the underlying topics. Motivated by the above observation, this thesis proposes two approaches to counter the data sparsity and under-specificity in tweets for topic modeling tasks: i) expanding tweets with semantically related hashtags, and ii) prioritization of selected hashtags. From various experimental results, it is evident that our proposed methods enhance the topic modeling performance either by i) tweet expansion with semantically related hashtags or ii) incorporating prioritized hashtags in LDA. Furthermore, this thesis investigates the effect of LDA in relation prediction as a case study by exploiting topic and entity relation. It is observed that event-centric relations are effectively predicted using topic modeling over news articles.

Contents

List of Figures

List of Tables

Abbreviations

LDA	Latent Dirichlet Allocation
BTM	Biterm Topic Model
HP-LDA	Hashtag Prioritized LDA
PNE-LDA	Prioritized Named Entity driven LDA
GTD	Global Terror Data
LSA	Latent Semantic Analysis
NMF	Non-negative Matrix Factorization
PLSA	Probabilistic Latent Semantic Analysis
TF-IDF	Term-Frequency Inverse Document Frequency
TF	Term Frequency
EM	Expectation Maximization
AT	Author Topic
CAT	Citation Author Topic
ART	Author Recipient Topic
TOT	Topics over Time
DTM	Dynamic Topic Model
cDTm	continuous time Dynamic Topic Models
L-LDA	Labeled LDA
sLDA	Supervised LDA
Seeded-LDA	Seeded-LDA
Source-LDA	Source-LDA
TimeUser LDA	TimeUser LDA
TimeReliableUser LDA	TimeReliableUser LDA
Twitter-LDA	Twitter-LDA
HGTM	Hashtag Graph based Topic Model
MGe-LDA	hashtag-based Mutually Generative LDA
GCN	Graph Convolution Network
ESA	Explicit Semantic Analysis
WTMF-G	Weighted Textual Matrix Factorization Graph

IR	Information Retrieval
TSTM	Topic Specific Translation Model
TTM	Topic Translation Model
FFNN	Feed forward Neural Network
BiLSTM	Bidirectional Long Short Term Memory
BERT	Bidirectional Encoder Representations from Transformers
GSTN	Goods and Services Tax Network
CAB	Citizenship Amendment Bill
NMI	Normalized Mutual Information
JC	Jaccard Coefficient
betC	betweenness centrality
closC	closeness centrality Centrality
prC	page rank centrality
NMI	Normalized Mututal Information
JC	Jaccard Coefficient
SNA	Social Network Analysis
AUC	Area under ROC curve
ETM	Embedded Topic Model
CN	Common Neighbor
AA	Adamic-Adar Index
RA	Resource Allocation

Chapter 1

Introduction

Topic modeling is a statistical tool to find latent topics or themes from unstructured data. Topic modeling has gained significant importance in last decades and has been applied over various domains such as texts [3, 4, 5], images [6, 7, 8], and music collections [9, 10]. Topic modeling helps users understand large data collections by annotating them with a topic, presenting its summarized view, and providing tools for interactive searching. Thereafter, these latent topics can be used for various tasks such as document clustering [11, 12], information retrieval [13], sentiment analysis [14], and recommendation systems [15]. Topic modeling has been applied to different text datasets such as news publications, Wikipedia pages, blogs, customer's reviews, and research publications for text mining [4, 5, 11, 13, 16], sentiment analysis [14], content recommendation [17, 18], word sense disambiguation [19], and mining research interest of an author [20, 21].

Topic modeling has been exploited in many online social media platforms. Social media platforms disseminate vast amounts of information, opinion on world events like elections [22, 23], pandemic [24], movie releases [25], and marketing of new products [26, 27]. Twitter, one of the most popular social media platform with a character limit of 280 in each post [1], has 192 million daily active users, 500 million tweets sent per day [2]. Twitter has become a prominent platform of information dissemination for various activities such as protests [28], attacks [29], natural disasters [30], movie releases [25], elections [22, 23], and pandemic [24]. Due to the massive volume of information continuously generated about diverse topics, Topic modeling on Twitter has become a challenging and interesting problem. Latent Dirichlet Allocation *(LDA)* [2] is a widely used topic modeling method

[1] https://developer.twitter.com/en/docs/counting-characters
[2] https://www.oberlo.in/blog/twitter-statistics

for discovering latent topics inherently present in documents collection. LDA uses word to word co-occurrences in a document to create topics representation from the text collection. In case of short-length documents, word to word matrix becomes sparse and degrades the performance of LDA [31, 32]. In case of tweets, this sparsity further increases due to diverse nature of used texts such as multilingual, code-switching, misspelling, short text, elongated text, emojis, mentions, and hashtags [32].

In the past, researchers have tried to solve the problem of sparsity and under specificity by augmenting tweets with the content from external sources such as news media, Wikipedia pages, web pages, and the content of URL's present in the tweet [33, 34]. Given a short and noisy text, discovering related text from external sources is a non-trivial task, considering the diverse nature of the text present in a tweet. Another way of solving the data sparsity and under specificity is by adding related tweets to a larger pseudo document and applying LDA over the expanded document. Different ways of combining related tweets such as hashtag-centric [35, 36], user-centric [31, 37], and communication centric [38] have been studied. Though combining related tweets improves the performance of LDA over raw tweets, the distribution of topics over the individual tweet may get lost, and the size of the corpus may also increase due to the presence of duplicate tweets in a different larger pseudo document. Other possible ways to address the above problem are using lexical normalization [39], utilization of temporal information, incorporating user weights in tweet network [40], and bigram-based Biterm Topic Model *(BTM)* [41]. However, all the additional information used in the above studies are not always available, especially with the publicly available datasets. For example, only a small percentage of tweets are geotagged [42] and predicting the location of non geotagged tweets is a challenging problem. Similarly, collecting all the tweets of a user to estimate users' topic distribution is limited by Twitter API rate limit[3].

In the studies [43, 44, 45], hashtags are found to provide important meta information in linking tweets with underlying topics. Hashtags are created by the users who posted the tweet, providing explicit reference to the underlying topics. Our preliminary study also shows that hashtags play a significant role in finding topics of the tweets (a brief study will be found in Section 3.5.1). Motivated by the above study, this thesis is focused on leveraging the importance of selected tokens such

[3] https://developer.twitter.com/en/docs/twitter-api/v1/tweets/timelines/overview

as hashtags to improve topic modeling performance in case of short, noisy and under specified tweets.

1.1 Challenges

Topic modeling on Twitter has numerous challenges, making it a difficult task. Table 1.1 shows few examples of noisy and under specified tweets. The key challenges for the topic modeling in tweets are listed below:

- **Short and Noisy text**: The performance of LDA on regular text with formal writing is considerably good. However, due to character limit of 280 in tweet[4], user often write in informal language and irregular way. Short and irregular text degrade the performance of LDA.
- **Under specified text**: Many times, a tweet contains only partial information about an event, and the other information is either encapsulated in the hashtags, external link or even not present in the tweet. Gao et al. [46] shows that Twitter post have medium specificity. The mean score of specificity out of 7000 manually labelled tweets is 2.64 on the scale of 1 (Very General) to 5 (Extremely Specific), and more than 75% of tweets have a specificity score less than 3.16. The tweets with specificity score 4 and 5 have dominating explicit references to people, objects, and events as compared to the textual contents.
- **Creative writing and misspelling**: The user tends to write diverse words forms and often misspelled words in a tweet. Many times, two or more words are written as a single word using camel case writing. The word normalization of a tweet is itself a challenging task.
- **Data sparsity**: Due to the limited number of word in tweets and non-uniform way of writing, i.e., shortened forms, elongated forms, misspelling; word to word co-occurrence at the tweet level is often sparse. This leads to challenges in converging proper topics in case of LDA.
- **Multilingual and code-switching**: The tweet is written in multilingual content. Sometimes within a tweet there are switching between languages.

[4] https://developer.twitter.com/en/docs/counting-characters

S.No	Tweet
1	@narendramodi Congratulations for passges GST in LokSabha..
2	@PMOIndia URL
3	@ArvindKejriwal Pakbggstbenficiryofur dmnd askng fr prof of srgicl striks.Nxtdy Kjriwlappersonfrnt pgeofal Paknwspaprs. Wht's the deal,MrCM
4	@BDUTT burhan wali azaadi nahi chahiye ??? URL

TABLE 1.1: Short, noisy and under specified tweet example

Furthermore, tweets are also written in multiple languages using transliteration. This further increases the challenge of capturing different words with similar meaning, and converge to good topics.

- **Shared vocabulary across different topics**: In tweets, the people often tag celebrities, news reporters, and political leaders to maximize the reach of their tweets. Moreover, one event leads to other events, causing a shared vocabulary of words, hashtags, and user mentions across different topics. This causes LDA to give a combined representation of two topics into a single topic.

1.2 Objectives and Scope of the Thesis

The objectives of the thesis are to exploit the benefits of hashtags to address the problem of data sparsity and under-specificity in tweets. It proposes the following two approaches to handle data sparsity and under-specificity of tweets:

- **Expand the tweets with related hashtags**: Expand tweets with semantically related hashtags to address the problem of data sparsity and under specificity.
- **Prioritize the important tokens (hashtags)**: Assign different importance to different tokens in the corpus and guide the topic discovering process using the priority of the token.

Though different topic modeling approaches such as Latent Semantic Analysis *(LSA)*, Non-negative Matrix Factorization *(NMF)*, Probabilistic Latent Semantic Analysis *(PLSA)*, Latent Dirichlet Allocation *(LDA)* have been discussed in the literature, this thesis focuses extensively on LDA for the above-mentioned objectives.

1.3 Contributions Made in the Thesis

1.3.1 Significance of Hashtags in discovering topics

As reported in studies [43, 44, 45], hashtags play a significant role in grouping tweets of related topics. This thesis investigates the influence of hashtags in LDA by performing following analyses; i) perform LDA over raw tweets, ii) remove hashtags from the raw tweets and perform LDA, iii) remove mentions from the raw tweet and performs LDA, and iv) remove hashtags and mentions from the raw tweet and perform LDA. From various experimental setups over different datasets, we found that hashtags plays a significant role in the performance of LDA. In the next two contributions, we propose to utilize the hashtags for improved topic modeling over tweets.

1.3.2 Hashtag based tweet expansion for improved topic modeling

As discussed in section 1.1, LDA performance degrades in case of tweets due to short and under-specified texts, as it fails to mine the proper context. In this contribution, we propose to expand tweets by adding related hashtags using text-based and network-based approaches. This solves the problem of data sparsity and under-specificity by increasing the number of words in tweets, adding meaningful context in terms of hashtags. We systematically expand tweets by adding different number of hashtags (2, 4, 6, 8, 10) and compare the performance of LDA over expanded tweets and raw tweets.

1.3.3 Prioritizing hashtags for improved topic modeling

In this contribution, we propose a variant of LDA named as Hashtag Prioritized LDA *(HP-LDA)* which incorporates weight of each token. Specially, we want to give higher weights to hashtags while discovering topics. We systematically evaluated different hashtag prioritization strategies: a) prioritize all the hashtags, b) prioritize prominent hashtags, i.e., network centrality, c) prioritize manually identified hashtags based on domain knowledge. We compare the performance of HP-LDA with LDA and other non-prioritized counterparts.

To further investigate the effect of topic modeling using prioritized tokens in news collection, we extend HP-LDA with prioritizing named entities to find topics in highly overlapping clusters. From experimental observations over three different datasets of different nature (i.e., Bomb Blast, Reuters-21578, and 20-Newsgroup), it is evident that our proposed *Prioritized Named Entity driven LDA* (PNE-LDA) outperforms its LDA counterparts for entity-driven topics in terms of F-measure and Rand Index.

1.3.4 Application of LDA in relation prediction

Application of LDA in various downstream tasks have been reported in several studies such as research article reviewers suggestions [20, 21], content recommendation [17, 18], and word sense disambiguation [19]. Motivated by such studies, this thesis has also conducted a special study in predicting terror attack using LDA. We attempt to predict future terror attack by discovering potential future relationships between different attack related attributes. For this study, we use Global Terrorist Data *(GTD)* [5] and corresponding news reporting. Each document in GTD is a short description of one terrorist attack with the attributes such as *country, region, province, city, attack types, attack subtypes, organization involved, weapons used etc.*. For each terror attack, we crawl corresponding news articles given in the GTD dataset. Each row of GTD corresponds to an event and is treated as a document along with the content of associated news articles. It is evident that relation prediction can be improved by incorporating latent topics discovered from news publications. However, influence of latent topics (discovered using the proposed enhanced LDA over tweet collection) in predicting various relationships in tweets such as hashtags-to-hashtags, mention-to-mentions, hashtags-to-mention, hashtags-to-users, etc. has not been included in the thesis. It can be explored as a future work following this thesis work.

1.4 Organization of the Thesis

The remaining part of the thesis is organized as follows:

- **Chapter 2 Literature Review**: This chapter briefly reviews different Latent Dirichlet Allocation *(LDA)* based topic modeling approaches in regular

[5] http://www.start.umd.edu/gtd/

and short text, and also briefly highlight the different approaches of handling under-specificity in short and noisy tweets.

- **Chapter 3 Hashtag based tweet expansion for improved topic modeling**: In this chapter, we investigate the significance of hashtag for topic modeling in tweets. For expansion of tweets with semantically related hashtags, we empirically evaluate different hashtag prediction/recommendation methodology. We propose the method of tweet expansion to counter under-specificity and investigate its effect on topic modeling.

- **Chapter 4 Prioritizing Hashtag for improved topic modeling** : Given a text corpus, different tokens may have different level of influences in discovering latent topics. LDA exploits frequency and co-occurrence characteristics between tokens. Apart from the frequency, influence of a token may also be defined by external factors. This chapter proposes a variant of LDA which incorporates explicitly supplied influence of a token, and guide the topic modeling process.

- **Downstream Application of LDA – A case study on terror attack prediction**: LDA has been used in various downstream applications such as research article reviewers suggestions, content recommendation, word sense disambiguation. This section presents a case study of using LDA in relation prediction between attributes of a terror attack.

- **Chapter 6 Conclusion and Future Work**: This chapter presents conclusion of the thesis with few possible future research directions to the thesis.

Chapter 2

Background Studies

Topic modeling is a well explored area of study which aims at discovering latent representation in a large text corpus [12, 47, 48]. Topic models help in organizing, searching, indexing, and interactive browsing of large collections. With the invention of new digital platforms, novel benchmark arises to cope with different forms of data using with Topic modeling Latent Dirichlet allocation *(LDA)* is one of the most explored topic modeling techniques and therefore, the proposed objective exploits mainly the same in this thesis. This chapter briefly presents different LDA-based topic modeling techniques and response of LDA over noisy and under-specified texts.

2.1 Topic Modeling

Topic modeling refers to a set of models/algorithms that are used to organize, search and find similar documents from a large collection of documents. Generally, topic models represent each document as a mixture of topics and topics as a mixture of words. Topic modeling approaches can be broadly grouped into two categories, a) Matrix Factorization-based approaches, b) Probabilistic approaches. Latent Semantic Analysis *(LSA)* [49] and Non-negative Matrix Factorization *(NMF)* [50] are matrix factorization-based topic models. Further, Probabilistic Latent Semantic Analysis *PLSA* [51] and Latent Dirichlet Allocation *(LDA)* [2] are probabilistic topic models.

Latent Semantic Analysis (LSA): Traditionally, a large text collection with D documents and W unique words in the vocabulary is represented by a word-document count matrix X_{W*D}, where each column represents a document vector, and each row represents the unique word of the vocabulary. The entries in the X matrix usually corresponds to either term frequency score or TF-IDF score. The high dimension of document vector affects the performance of different text mining tasks such as searching, indexing, and classification. Latent Semantic Analysis *(LSA)* [49] solves the problem of high dimension document vector by representing it in terms of latent space instead of vocabulary size. LSA uses Singular Value Decomposition *(SVD)* to reduce the dimensions of document vector from W to K latent topics. At first LSA normalizes the word-document count matrix X to $\tilde{X}$ and then, secondly, factorize the matrix as a product of three matrices as follows:

$$X_{W*D} \approx \tilde{X}_{W*D} = U_{W*R}\ S_{R*R}\ (V_{D*R})^T \approx\ U_{W*K}\ S_{K*K}\ (V_{D*K})^T$$

where, R is the rank of document-to-document similarity matrix ($\tilde{X}^T * \tilde{X}$), K is the number of dimension in latent space (topics), and the columns of U and V are orthogonal vectors. Each row of the matrix U is the eigen vector of word-to-word similarity matrix ($\tilde{X} * \tilde{X}^T$), and presents a word in the R latent dimensions. Similarly, each column of V^T is the eigen vector of document-to-document similarity matrix ($\tilde{X}^T * \tilde{X}$), and presents a document in R latent dimension. The diagonal matrix S contains the square roots of non-zero eigen values of ($\tilde{X}^T * \tilde{X}$) in descending order. The dimension of the latent space is further reduced from R to K by truncating the last $(R-K)$ rows of U, $(R-K)$ columns of V^T, and $(R-K)$ rows and $(R-K)$ columns of the matrix S. The truncated rows of U and columns of V^T corresponds to lower eigen values, and are less informative. The columns of V^T represent documents using K latent topics, and the rows of U represent words of vocabulary in K latent topics. LSA's word-count factorization can produce implausible findings, such as negative counts, due to its assumption of a Gaussian noise model.

Non-negative Matrix Factorization: Non-negative matrix factorization *(NMF)* [50] addresses the shortcomings of LSA by restricting the decomposition of the document-word count matrix to the product of a pair of non-negative matrices. The NMF decomposition of word count matrix X is given by

$$X_{W*D} \approx \tilde{X}_{W*D} = U_{W*K}\ V_{K*D}$$

where U and V are constrained to be non-negative matrices and K is chosen

smaller than W and D. The rows of U represent words in K latent space (topics), and columns of V represent documents in K latent space (topics). Frobenius norm and Kullback-Leibler divergence are two widely used cost functions with NMF for minimizing the distance between $\tilde{X}$ and the product of U, V matrix $(U * V)$. Unlike LSA, NMF ensures that entries of word-topic matrix (U) and topic-document matrix (V) are non-negative values only.

Probabilistic Latent Semantic Analysis (PLSA): Hoffman proposed a probabilistic model named as Probabilistic Latent Semantic Analysis *(PLSA)* [51] as an alternative to LSA. In PLSA, documents are represented as a mixture of T latent topics, and each latent topic is represented as a Multinomial of distribution over words in the corpus. To generate a word in a document, a topic is first generated from the document-specific mixture of topics and thereafter, a word is generated using the Multinomial distribution associated with that topic. Therefore, each word is generated from a single topic and thereby, words in the same document can be generated by multiple topics. The conditional probability of a word w in a document d under PLSA is given by:

$$p(w|d) = \sum_{t} p(w|t)p(t|d)$$

where $p(w|t)$ represents the probability of word w under the Multinomial distribution associated with topic t and $p(t|d)$ represents the probability of topic t under the Multinomial distribution associated with document d.

In PLSA, the mixture of latent topics associated with each document are treated as parameters of the model instead of random variables generated from a higher level process. Given the words in the corpus, PLSA estimated the word-topic $p(w|t)$ and topic-document $p(t|d)$ parameters using the Expectation Maximization *(EM)* algorithm.

2.2 LDA

LDA [2] is a widely used probabilistic graphical model for finding latent semantic topic distribution in a document collection. LDA extends PLSA by incorporating Dirichlet prior over document-topic and topic-word distributions. Similar to PLSA, LDA models each document as multinomial distribution over topics, where

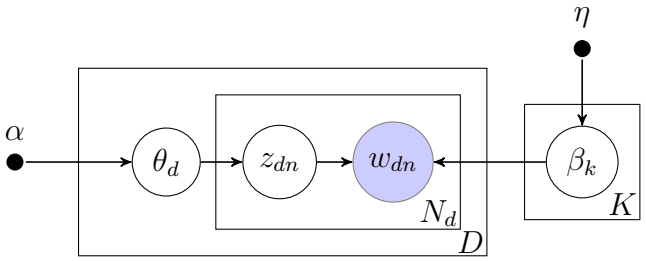


FIGURE 2.1: Plate Diagram of LDA. [1]

ALGORITHM 1: The generation process of LDA [2].

```
// Topic Word Generation
for each topic k in K : do
    Generate topic word distribution β_k ~ Dir(η)
end
// Generation of each word of every Document one by one
for each doc d in Doc : do
    Choose θ_d ~ Dir(α)
    for each of the word in doc d : do
        Choose a topic Z_nd ~ Multinomial(θ)
        Choose a word w_nd ~ Multinomial(β_zdn)
    end
end
```

each topic is further expressed as multinomial distribution over words in the vocabulary. However, unlike PLSA, the document-topic and topic-word multinomial distributions in LDA are treated as random variables and samples from Dirichlet distributions. Figure 2.1 presents a graphical plate model of the LDA encoding relationship between different random variables. $\theta_{(D*K)}$ is a matrix representing document-topic multinomial distribution and θ_d corresponds to the topic distribution of the d^{th} document, where D is the total number of documents in the corpus, $d \in [1, D]$, and K is the number of topics. Similarly, $\beta_{(K*V)}$ is a matrix representing the topic-word multinomial distribution, where K represents the total number of topics and V represents the total number of words in the vocabulary. β_k corresponds to the word distribution of k^{th} topic, where $k \in [1, K]$. A document (d) has total N_d words, and z_{dn} represents the topic label assigned to a word w_{dn} (n^{th} word of d^{th} document). In the generative process of LDA, as given in the algorithm 1, we first sample topic-word distribution β_k using Dirichlet prior η for every topic $k \in [1, K]$. Further, each document is processed one by one. For every document, first document-topic distribution θ_d is sampled using Dirichlet prior α. For every word in the d^{th} document, a topic z_{dn} using θ_d and a word using topic-word distribution $\beta_{z_{dn}}$ are sampled.

The key posterior distribution to estimate the hidden variables of LDA (θ, $\mathbf{z}$) given a document vector ($\mathbf{w}$) can be written as follows [2]:

$$p(\theta, \mathbf{z}|\mathbf{w}, \alpha, \beta) = \frac{p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta)}{p(\mathbf{w}|\alpha, \beta)} \tag{2.1}$$

where the joint distribution $p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta)$ and marginal distribution $p(\mathbf{w}|\alpha, \beta)$ are defined as follows:

$$p(\theta, \mathbf{z}, \mathbf{w}|\alpha, \beta) = \prod_{d}(p(\theta_d|\alpha)\prod_{n} p(z_{d,n}|\theta_d)p(w_{d,n}|\beta, z_{d,n}) \tag{2.2}$$

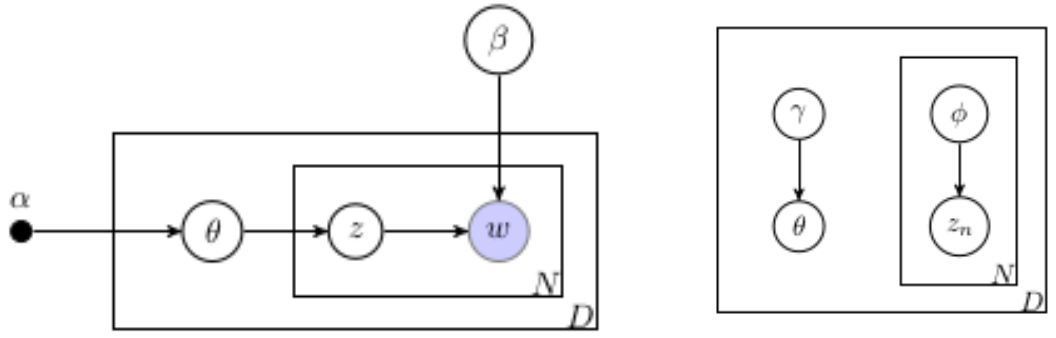
$$p(\mathbf{w}|\alpha, \beta) = \frac{\Gamma(\sum_i \alpha_i)}{\prod \Gamma(\alpha_i)} \int (\prod_{i=1}^{k} \theta_i^{\alpha_i - 1})(\prod_{n=1}^{N}\sum_{i=1}^{k}\prod_{j=1}^{V}(\theta_i \beta_{ij})^{w_n^j}) d\theta \tag{2.3}$$

The denominator (equation 2.3) in equation 2.1 is intractable due to the coupling between θ and β [2], making the computation of the posterior inference intractable. This is a classical problem of Bayesian inference, which is solved using either i) sampling-based approaches, or ii) approximation-based approaches. Gibbs sampling-based Markov Chain Monte Carlo *(MCMC)* [52, 12], and Variational Inference (mean field) [2] are commonly used sampling and approximation approaches for estimating parameters of LDA. We briefly discuss the Variational Inference and collapsed-Gibbs sampling in the subsections below.

2.2.1 Parameter estimation using VI in LDA

Variational inference is a deterministic approach to approximate the intractable posterior in case of Bayesian Inference. Given the observed data(X), hidden variables (H), and variational parameters (V); the intractable posterior $p(H|X)$ then can be estimated with a tractable distribution $q(H|X, V)$, where $q(H|X, V)$ is from a family of simpler distributions defined by a set of free variational parameters V. In the variational Inference, we find those parameters V which minimize the Kullback-Leibler KL divergence $KL(q(H|D, V)||p(H|D))$ to the true posterior. The KL divergence distance between approximate distribution q and true posterior p can be written as follows:

$$D_{KL}(p||q) = \sum_i p(i) log(\frac{p(i)}{q(i)}) \tag{2.4}$$



(A) Simplified LDA plate diagram.

(B) Decoupling of LDA plate diagram using VI.

FIGURE 2.2: Graphical representation decoupling in Variational Inference (VI) of LDA posterior approximation [2].

Blei et al. [2], decouple Bayesian inference of LDA and introduce variational inference parameters(γ, ϕ), as shown in figure 2.2. To approximate the posterior of the LDA, we define a tractable distribution $q(\theta, z|\gamma, \phi)$ as follows:

$$q(\theta, z|\gamma, \phi) = q(\theta|\gamma) \prod_{n=1}^{N} q(z_n|\phi_n) \tag{2.5}$$

where γ and ϕ are two sets of variational parameters, γ represents document-specific topic Dirichlet and ϕ represents word-specific topic multinomial. The probability of topic z given document d is given by $q(\theta_d|\gamma_d)$, where each document d has its Dirichlet prior over topics γ_d. Similarly, the probability of topic assignment to word $w_{d,n}$ is given by $q(z_{d,n}|\phi_{d,n})$, where each word $w_{d,n}$ has its multinomial over topics $\phi_{d,n}$. Inference is then performed by minimizing the Kullback-Leibler (KL) divergence between the variational distributions $q(\theta, z|\gamma, \phi)$ and the true posteriors $p(\theta, z|w, \alpha, \beta)$. The variational parameter $(\gamma,\ \psi)$ can be approximated using the equation below:

$$(\gamma^*, \psi^*) = \arg\min_{(\gamma^*, \psi^*)} D\big(q(\theta, z, |\gamma, \psi)||p(\theta, z|w, \alpha, \beta\big) \tag{2.6}$$

If we simplify the above equation, the objective function of Variational Inference of LDA can be defined as follows:

$$\begin{aligned} L(\gamma, \phi; \alpha, \beta) \quad &= \mathbb{E}_q\Big[log\big(p(\theta|\alpha)\big)\Big] + \mathbb{E}_q\Big[log\big(p(z|\theta)\big)\Big] + \mathbb{E}_q\Big[log\big(p(w|z, \beta)\big)\Big] \\ &\quad - \mathbb{E}_q\Big[log\big(q(\theta)\big)\Big] - \mathbb{E}_q\Big[log\big(q(z)\big)\Big] \end{aligned} \tag{2.7}$$

Blei et al. [2] propose a variational expectation maximization (variational EM) for

ALGORITHM 2: A Variational Inference Algorithm for LDA[2].

Initialize $\phi_{ni}^0 := \frac{1}{K}$ $\forall$ i and n. ;
Initialize $\gamma_i := \alpha_i + \frac{N}{K}$ $\forall$ i ;
while *convergence criteria is reached* **do**
    // At each t^{th} step
    for *n in range(1,N)* **do**
        for *i in range(1,K):* **do**
            Set $\phi_{ni}^{t+1} = \beta_{iw_n}\ exp\big(\psi(\gamma_i^t)\big)$;
        end
        Normalize ϕ_n^{t+1} to sum to 1 ;
    end
    Set $\gamma^{t+1} := \alpha + \sum_{n=1}^{N} \phi_n^{t+1}$
end

estimating the parameters, as shown in algorithm 2. For details, reader may refer to the paper [2].

2.2.2 Parameter estimation using collapsed Gibbs Sampling in LDA

Gibbs sampling algorithm is a MCMC based statistical algorithm, named after physicist Josiah Willard Gibbs, and was described by Stuart and Donald Geman in 1984. The basic version of Gibbs sampling can be considered as a special case of Metropolis-hasting MCMC sampling approach. Gibbs sampling is useful to sample the distribution, where the joint distribution is not known completely or difficult to sample from directly, but the conditional distribution of each variable is known and easy to sample from. Gibbs sampling is useful in the scenario, where the joint distribution $p(X_1, X_2, ..., X_n)$ is very complex or intractable but the conditional distribution, such as $p(X_1|X_2, X_3, ..., X_n)$, $p(X_2|X_1, X_3, X_4, ..., X_n)$ are tractable and easy to sample. The algorithm 3 presents a basic version of Gibbs sampling algorithm using MCMC approach.

This thesis uses we use collapsed Gibbs sampling [1] (a variation of Gibbs sampling) to estimate the parameters of LDA. The inference equation to estimate parameters of LDA, using collapsed Gibbs sampling, can be written as follows:

$$P(z_{dn} = j|z_{\neg dn}, w_{\neg dn}) \propto (\alpha + n_{\neg dn,j}^{(d)}) \frac{\eta + n_{\neg dn,j}^{(w_{dn})}}{V.\eta + n_{\neg dn,j}^{(.)}} \tag{2.8}$$

ALGORITHM 3: Gibbs Sampling with MCMC algorithm.

Input: X : set of all observed variables, i.e., $(X_1, X_2, ..., X_n)$
Output: θ : set of all non-observed variables i.e., $(\theta_1, \theta_2, ..., \theta_d)$

```
Fix the value of observed variables X ;
Initialize the non-observed variables randomly θ ;
Perform a random walk through a space of complete variable assignment;
while P(θ) keeps on changing do
    for each move do
        pick a variable θ_i;
        // Re-sample the chosen variable θ_i, keeping all (d-1)
           variable same.
        Calculate t = P(θ_i = True|θ_¬i, X);
        Set P(θ_i = True) = t
    end
end
```

TABLE 2.1: LDA variants parameter explanation

Name	Symbol	Details
z_{dn}	z_{dn}	Topic assigned to n^{th} word of d^{th} document
w_{dn}	w_{dn}	n^{th} word of d^{th} document
$z_{\neg dn}$	$z_{\neg dn}$	All topics-word assignment except the current word topic assignment
$w_{\neg dn}$	$w_{\neg dn}$	All words in the vocabulary except the current word
nmz	$n^{(d)}_{\neg dn,j}$	Number of words of current document assigned to current topic j except the current word w_{dn}
nzt	$n^{(w_{dn})}_{\neg dn,j}$	Number of word assigned to assigned to current topic j and similar to current word, except current word w_{dn}
nz	$n^{(\cdot)}_{\neg dn,j} = \sum_{\forall w_{dn} \in V} n^{(w_{dn})}_{\neg dn,j}$	Number of words assigned to current topic j except current word w_{dn}

Table 2.1 explains different symbols used in the equation 2.8. Detail derivation of the above equations is present in the study [53]. The first term of the equation 2.8 characterizes the probability of choosing a topic j, whereas the second term corresponds to the probability of word w_{dn} being assigned to the topic j.

2.3 Evaluation of Topic Modeling over document collection

Traditionally, most of the research papers [2] evaluated the performance of LDA by using it for clustering or classification tasks, and presented the top-words for

each topic. In this thesis, we have used majorly clustering criteria such as F-measure, Normalized Mutual Information *(NMI)*, Rand Index *(RI)*, and Jaccard Coefficient *(JC)* to evaluate the performance of different topic models. As given in the book [54] and webpage [1], the contingency matrix can be defined as the table 2.2.

TABLE 2.2: Contingency table for extrinsic clustering performance measure.

	Same topic	Different topic
Same class	TP	FN
Different class	FP	TN

Suppose we N number of documents. Then we compute the True Positive *TP)*, True Negative *(TN)*, False Positive *(FP)*, and False Negative *(FN)* by assigning each document pair to one of the cells of the contingency table 2.2. After finding TP, TN, FP, and FN, the different evaluation metrics can be defined as follows as per equation 2.9

$$\begin{aligned} Precision(P) &= \frac{TP}{(TP+FP)} \\ Recall(R) &= \frac{TP}{(TP+FN)} \\ \text{F-measure} &= \frac{2*P*R}{(P+R)} \\ \text{Rand Index} &= \frac{TP+TN}{(TP+TN+FP+FN)} \\ \text{Jaccard Coefficient} &= \frac{TP}{(TP+FP+FN)} \end{aligned} \tag{2.9}$$

Further, we have considered Normalized Mutual Information *NMI)* to evaluate the quality of clustering externally. Suppose, $T = \{t_1, t2_2, \ldots, t_K\}$is the set of topics obtained and $\mathbb{C} = \{c_1, c_2, \ldots, c_J\}$ is the set of classes. Normalized Mutual Information (NMI) can then be defined as per equation 2.10

$$\text{NMI}(T, \mathbb{C}) = \frac{I(T;\mathbb{C})}{[H(T)+H(\mathbb{C})]/2} \tag{2.10}$$

[1] Clustering evaluation

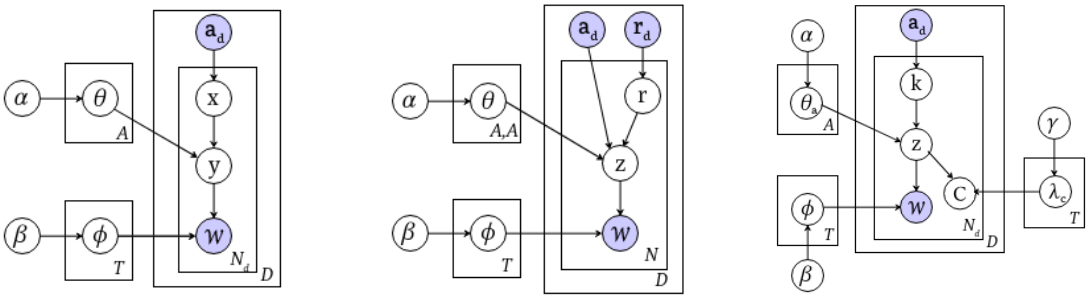
where $I(T;\mathbb{C})$ denotes the point wise mutual information between topic set (T) and class set $\mathbb{C}$, and H(T) and $H(\mathbb{C})$ denotes the entropy of the topics and classes respectively.

The advantage of Rand Index as evaluation criteria is that it penalizes both the false negatives and false positives in clustering. However, the numerator of rand index (equation 2.9) is dominated by true negative. The Jaccard Coefficient, in case of clustering evaluation, address this issue by removing true negative from both numerator and denominator. The F-measures supports a differential weighting scheme for the two types of errors (false negative and false positive). Further, normalized mutual information metric can be interpreted using information theory. In addition to clustering based evaluation metrics, we have also used topic coherence to measure the quality of the topics, as described in the study [55, 56].

2.4 Topic Modeling in regular text

Different variants of Topic have been proposed to model the contents from regular text with formal writing styles such as research papers, emails, news articles, web blogs and Wikipedia pages. This section describes few of topic modeling variants in regular text to include authors information, temporal information to improve the topic modeling performance, and its use case in different applications. Further, it also describes few of the supervised topic models to include labels, and other information.

Mining the research paper and email corpora: Author Topic *(AT)* model [20], Citation Author Topic *(CAT)* model [21], Author Recipient Topic *(ART)* model [21] are a few early works extending LDA to model the topic of the documents, author interest, and role from research publication and email corpora. Rosen et al. [20] proposed an Author Topic *(AT)* model, a variant of LDA, to model the authors' research interest and contents from the research paper collections. LDA assumes that each document d is associated with a distribution over topics (θ_d) whereas AT assumes each author a in A to have a distribution over topics (θ_a). Similar to LDA, each topic of AT is expressed as a distribution over words in the vocabulary. The graphical plate diagram of AT is given in Figure 2.3a. The generative process of AT is as follows. A group of authors a_d writes a document d. For each word in the document d, an author a is sampled uniformly from a_d. Further, a topic z is sampled from the distribution of topics specifics to that author (θ_a), and a word is generated using the topic-word distribution (β_z) of the selected topic. AT offers



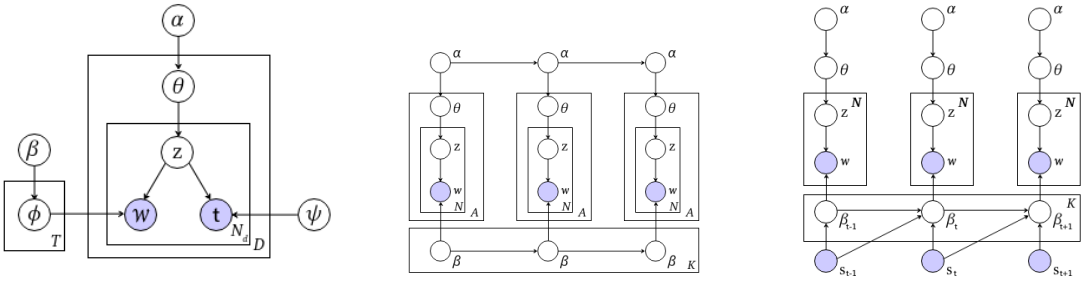
(A) AT [20]. (B) ART [21]. (C) CAT [57].

FIGURE 2.3: Plate diagram of Author Topic (AT) model, Author Recipient Topic (ART) model and Citation Author Topic (CAT) model.

the top ten words for each topic and the top ten authors to generate the word conditioned on the topic over full papers from NIPS conference and abstracts on CiteSeer. Representative topics corresponding to EM and mixture models, handwritten character recognition, reinforcement learning, SVM and kernel methods, speech recognition, and Bayesian learning were discovered using AT model over NIPS dataset, whereas topics corresponding to speech recognition, Bayesian learning, user interfaces, solar astrophysics were discovered by AT model over CiteSeer dataset. The model can find automatic reviewer suggestions for a given publication and find researchers with similar research interest.

McCallum et al. [21] extends Author Topic model to Author Recipient Topic *(ART)* model for topic mining, interaction relationship between sender and receiver, and people's roles, from Enron and academic email. Unlike research documents, emails have only one author but can have multiple recipients. ART, as shown in Figure 2.3b, models topic from each document conditioned over the pair of author and receiver. AT models the topic distribution of a document conditioned on an author, whereas ART models the topic distribution of a document conditioned on author and individual recipients. ART model represents each topic with top ten words with corresponding conditional probabilities and pairs of sender and receiver and corresponding probabilities. ART model can predict people roles and calculate similarities between people based on the role and topics. ART model outperforms AT in predicting peoples roles and similarities between their roles.

Yuancheng Tu et al. [57] extends the Author Topic model and Author Recipient Topic model to Citation Author Topic model *(CAT)* to mine the research communities from the research publication datasets grouped by the research interest and expertise. The CAT model, as shown in Figure 2.3c, includes citation information, where each topic is represented as a Multinomial distribution of overall cited



(A) TOT [58]. (B) DTM [59]. (C) cDTM [60].

FIGURE 2.4: Plate diagram of Topic over Time (TOT), Dynamic Topic Model (DTM) and continuous time Dynamic Topic Model (cDTM).

authors (λ_c). The generative process of a document using CAT is as follows. For each word of the document, CAT samples an author from observed Multinomial distribution, samples a topic (z) using Author-topic (θ_a) distribution, and samples a word and a citation using Topic-word (ϕ_t) and Citation-topic (λ_t) distribution respectively. CAT model can be used for authorship prediction, paper reviewer recommendation, research communities detection, and exploratory and interactive searching of research papers. For finding research experts corresponding to query words, a CAT-based retrieval system has better performance than AT in terms of Mean Average Precision.

Incorporating temporal factors to improve Topic Modeling: Another variants of LDA such as Topics over Time *(TOT)* [58], Dynamic Topic Model *(DTM)* [59], continuous time Dynamic Topic Models *(cDTM)* [60] incorporate temporal factors to find granular topics and evolution of topics over time from the text documents like scientific publication, news articles, and email corpus. Topics over Time (TOT) [58] models time and word occurrence together to form more granular and subtle topics. TOT represents the document time stamps by normalizing it between 0 and 1, and models the time distribution of each topic using a Beta distribution. The graphical model of TOT is shown in Figure 2.4a. For all the words in a document, the generative process of TOT first samples a topic using doc-topic Multinomial, and then samples a word and normalized timestamp for the chosen topic using topic-word Multinomial and Beta distribution as follows:

$$z_{dn}|\theta_d \sim Multinmoial(\theta_d)$$

$$w_{dn}|\phi_{z_{dn}} \sim Multinomial(\phi_{z_{dn}})$$

$$t_{dn}|\psi_{z_{dn}} \sim Beta(\psi_{z_{dn}})$$

where index dn refers to n^{th} word of d^{th} document, z_{dn}, t_{dn} are topics and timestamp associated with word w_{dn}. Few representative topics reported in papers from three datasets are as follows: i) Mexican war, Panama Canal, Cold War, Modern Tech from U.S. presidential address (1790-2002) dataset, ii) Faculty recruiting, ART paper, MALLET, CVS operations from 9 months (JAN-SEP 2004) of emails archive dataset of second author (Andrew McCallum), and iii) Recurrent NN, Game Theory from 17 years of NIPS proceedings (1987-2003) dataset. Topics discovered by TOT model contains more event specific words in top 10 words and are better localized in time of the events as compared to LDA topics. Further, average KL divergence between topic-word distributions in TOT is more than LDA, which implies LDA discovers more distinct topics. Beta distribution in TOT allows generation of more distinct topics by separating topics corresponding to events occurring at different time spans. Moreover, TOT have better capability of predicting timestamp (decade) of a word in the given document with 20% less L1 error, and twice accuracy score compared to LDA.

Dynamic Topic Model (DTM) [59] inculcates the evolution of topics over the years in scientific publications. DTM, as shown in Figure 2.4b, processes the document collection in sequential manner, overcoming the implicit assumption of document interchangeability of LDA. The documents are grouped by year, and each year documents is generated using the topics evolved from last year topics. DTM uses the state space model to capture the evolution of topics and topic proportion prior (α). The evolution of k^{th} topic at time slice t ($\beta_{t,k}$) and topic proportion prior (α_t) at time slice t, using parameters from previous timestamp $t-1$ is modelled using logistic normal distribution as below.

$$\beta_{t,k}|\beta_{t-1,k} \sim \mathcal{N}(\beta_{t-1,k}, \sigma^2, I)$$

$$\alpha_t|\alpha_{t-1} \sim \mathcal{N}(\alpha_{t-1}, \sigma^2, I)$$

The authors model the evolution of topics over 30000 articles published between 1881-1999 collected from JSTOR [2], and presented the change in top terms used in the corresponding topics over the year. Further, authors also evaluated the predictive power of DTM for predicting topics that will be published next year in the journal. Wang et al. [60] proposed continuous time Dynamic Topic Model ($cDTM$) extending classical DTM. DTM models the document collection using discrete time stamp for every year, whereas cDTM models the continuous time using Brownian motion [61]. The model diagram of cDTM is given in Figure 2.4c.

[2] https://www.jstor.org/

The predictive perplexity and time stamp prediction by cDTM over two news corpora were reported. Pan et al. [62] proposed SpaceTimeLDA to incorporate document publishing time information and location information into LDA to detect events from TDT3 [63] and Reuter news corpus with an intuition that different reporting of same event share location and temporal information.

Supervised Topic Models: To include the supervised information into LDA, Labeled LDA *(L-LDA)*[64] establishes one-to-one relationship between LDA topics and user defined class labels. Thereafter, several variants of LDA have been proposed to include the supervised information; for e.g, Supervised LDA *(sLDA)* [65], *DiscLDA* [66] and *MedLDA* [67] for classification of documents with single label; while *DP-MRM* [68], and *Dep-LDA* [69] and Boost Multi class L-LDA [70] for classification of documents with multiple labels. *Source-LDA* [71] has been proposed to include external information from Wikipedia to guide the topic word formation. Seeded-LDA [72] is one of the variants of LDA which incorporates the user's understanding of the corpus, and bias the topic formation process using representative word of each topic, which is more useful for the various extrinsic tasks such as document classification.

2.5 Topic Modeling in social media domain

In the past, several studies improved the topic modeling performance by utilizing the different meta information such as location information, tweet publishing time information, user's profile information, user's tweets and re-tweet count, tweet location, and bursty keyword information in case of tweets [73, 40]. Diao et al. [73] proposed a TimeUserLDA to detect bursty topics on the Twitter dataset by incorporating tweet posting time and user timeline activity information. Authors in [73] assume that each tweet contains only one topic, where topic distribution of a tweet is either dependent on user personal interest(local topics) or on timestamp (global topics). Similarly, a word is sampled for every tweet, either from a Multinomial of background words or topic-word distribution. The authors reported improvement in burst topic detection performance in terms of Precision@5 as compared to LDA, and other two variants of TimeUserLDA. Tsolmon et al. [40] proposed TimeReliableUser LDA to detect event incorporating the word weights based on time and user weights based on activity (weekly tweet and re-tweet count), and user popularity in tweet network. The authors reported improvement in event detection performance in terms of Precision over LDA [2] and TimeUserLDA [73] for

Korean tweets. Zhao et al. [32] proposed *(Twitter-LDA)* extending Author Topic model by assuming tweet to have a topic distribution over user and all the words of a tweet having a single topic. However, all the additional information used in the above studies are not always available, especially with the publicly available datasets. For example, only a small percentage of tweets are geotagged [42] and predicting the location of non geotagged tweets is a non-trivial problem. Similarly, collecting all the tweets of a user to estimate user topic distribution is limited by Twitter API rate limit[3].

Yan et al. in [74] proposed Bi-Term Topic Model *(BTM)* to handle document wise word sparsity in short text, by modeling a global corpus-specific topic distribution (*theta*) instead of modeling document-specific topic distribution θ_d. Further, BTM utilizes word co-occurrence pattern by sampling bi-terms instead of sampling an unigram for every document as in LDA. The paper reported improved performance of BTM as compared to LDA in terms of topic coherence and H-score (ratio of intra-cluster distance to inter-cluster distance) over Tweet-2011 used in TREC-2011 microblog task [4]. Wang et al. in [43, 75] proposed an extension of LDA named as Hashtag Graph-based Topic Model *(HGTM)* to handle short text tweet sparsity by harnessing the hashtag-hashtag relation based on tweet co-occurrences. The HGTM model assigns a hashtag and topic pair for every word of a tweet. The authors reported improved performance of HGTM over LDA and other topic models such as Author Topic Model *(ATM)*, Latent Semantic Analysis in terms of H-score over Tweet-2011 datasets. Xing et al. [76] proposed hashtag-based Mutually Generative LDA *(MGe-LDA)* for sub-event discovery in tweet collection utilizing the hashtags. In MGe-LDA, both hashtag and topic mutually generate each other to mine the relationship between hashtags and topics. The authors reported an improved H-score over three sub-event from Tweet-2011 collection in H-score compared to LDA, HGTM and Author Topic Model.

2.6 Significance of hashtags in social media

Hashtags have become an important feature across in short text social media platforms such as Twitter for searching, indexing and retrieval of related tweets. Hashtags often contain important keywords or entities related to the context of the tweets. Hashtags have been used in different text mining and natural language

[3] https://developer.twitter.com/en/docs/twitter-api/v1/tweets/timelines/overview

[4] https://trec.nist.gov/data/tweets/

processing task such as event detection [77, 78, 76, 43, 79, 80], topic detection [37, 81], sentiment analysis [77, 82], cross-platform information retrieval [83, 84], and personalized news and feed recommendation [85, 86]. The study [87] reported that mean persistence score (probability of passing of a piece of information) of hashtags in topics such as politics, sports, and technology is greater than 68%. The short text nature of the tweet with creative and informal writing style make *Topic detection* a non-trivial task. Motivated by earlier studies on utilizing hashtags on various application such topic detection, event detection, sentiment analysis, and persistence of hashtags in related tweets, we propose to study and utilize the effect of hashtags in improving Topic Modeling performance in tweets.

2.7 Summary

This chapter presents a brief description of Topic Modeling, and different topic models such as Latent Semantic Analysis *(LSA)*, Non-negative Matrix Factorization *(NMF)*, Probabilistic Latent Semantic Analysis *(PLSA),* and Latent Dirichlet Allocation *(LDA)*. It discusses the generative process and algorithm of LDA in detail, and evaluation metrics used to access the performance of topic models. Further, this chapter presents a few topic modeling studies used in regular text and social media domain. This chapter is concluded with an overview of significance of hashtags in various application over short texts such as Topic detection, Event detection, Sentiment analysis, and personalized news recommendation.

Chapter 3

Hashtag Based Tweet Expansion for Improved Topic Modeling

The preceding chapter briefly discusses the background studies of various topic modeling methods such as LDA and extensions of LDA used in regular and short texts. The performance of LDA and most of its extensions degrades over short and noisy tweets due to data sparsity and under-specificity. This chapter proposes to improve topic modeling performance over tweets by expanding with semantically related hashtags to counter data sparsity and under-specificity.

3.1 Introduction

Topic modeling on Twitter has become an important research problem as Twitter has become a prominent platform of information dissemination for various activities such as protests [28], attacks [88, 89, 29], natural disasters [30, 90, 91, 92], movie releases [93, 25], elections [22, 94, 23, 95], and pandemics [96, 97, 24]. Topic modeling provides a method of learning representative topics from a collection of documents. Discovering hidden topics inherently present in a text collection helps in various applications such as text summarization [98, 99, 16], document clustering [100, 11], text classification [101, 102, 4, 5], information retrieval [13, 103], and sentiment analysis [14].

Latent Dirichlet Allocation *(LDA)* [2] is one of the widely used topic modeling methods for finding topics from a document collection. Though LDA has shown promising results with well-formed text such as news articles, web pages, and

blogs; the performance degrades while dealing with short and noisy texts such as tweets [32]. Tweets are short in nature with a character limit of 280[1]. Tweets consist of texts with diverse nature such as multilingual, code-switching, misspelling, shorten text, elongated text, emojis, mentions and hashtags. While dealing with tweets, such diversity in the text often leads to data sparsity and under-specificity.

In the past, several studies have attempted to address the above problem of topic modeling in tweets, either by pooling tweets or text augmentation. In tweet pooling, related tweets are combined into a single pseudo-document (pool) and apply LDA over the expanded documents. Different pooling mechanisms such as hashtag-centric [35, 36], user-centric [31, 37], and communication-centric [38] have been studied. Though pooling-based tweet expansion methods improve the performance of LDA in comparison to un-pooled tweets, they only provide the topic distributions of a pool, but not for the individual tweet. Further, the corpus size may also increase due to the presence of the same tweet in multiple pools. In the tweet augmentation approach, tweets are expanded using relevant texts drawn from external sources such as news media, Wikipedia pages, and the URL present in the tweet [85, 33, 104, 34]. Given a short and noisy text, discovering related text from external sources is a challenging task, considering the diverse nature of the text present in a tweet.

From earlier studies [43, 44, 45], it is observed that the hashtags present in a tweet provide useful meta information linking the tweet to underlying topics or themes. These are created and embedded by the person who posted the tweet, providing manual references. Motivated by these observations, this chapter aims to address the problem of data sparsity and under-specificity by expanding tweets with semantically related hashtags drawn from a similar source (i.e., tweet collection). This chapter proposes two approaches utilizing textual content and network structure to discover semantically related hashtags for a given tweet. Given a tweet, the sequential models namely, BiLSTM [105, 106] and BERT [107] harness textual content to predict semantically related hashtags. Further, 1-hop neighbors and Graph Convolution Network *(GCN)* [108, 109] utilize the network structure of tweets to discover semantically related hashtags. After expanding the tweets with semantically related hashtags, the LDA is applied to discover the inherent topics. To study the strength of the proposed approaches and the importance of hashtags in topic modeling, LDA has been applied over various experimental setups such as original tweets, tweets after removing hashtags, and tweets after expanding with

[1] https://developer.twitter.com/en/docs/counting-characters

hashtags over two datasets of diverse nature (i.e., (i) Heterogeneous – tweets collected from dissimilar topics and (ii) Homogeneous – tweets collected from similar topics). From various experimental results, it is observed that the performance of LDA significantly improves after expanding tweets with semantically related hashtags.

3.1.1 Contributions

The key contributions of this chapter are as follows:

- Curated two real-world event related tweet datasets namely `Heterogeneous Dataset` and `Homogeneous Dataset` focused on popular events happened in India during 2016, 2019, and 2020. The Heterogeneous dataset consists of tweets related to diverse class of topics such as Attack, public response to government policies (CAB protest, GSTN) and parliament elections (BiharElection2020). And, The Homogeneous dataset consists of related class of topics such as Uri Attack, Pathankot Attack, Surgical Strike, Kashmir Unrest and Syria Crisis.
- Empirically evaluated the role of hashtags in the topic modeling performance over Heterogeneous and Homogeneous real-world event-based tweet dataset by checking different combination of different entity types such as hashtags, user mentions and general words.
- Proposed framework of hashtags-based tweet expansion using text-based based sequential (BiLSTM, BERT) and network-based graphical approaches (1-hop neighbor, GCN) (refer to Figure 3.1).
- Studied the effect of the proposed tweet expansion approaches by comparing LDA performance over raw tweets and expanded tweets on two datasets (Homogeneous and Heterogeneous) of different nature.
- Analyzed the topic quality produced by LDA over raw tweets and different tweets expansion approaches over the two datasets.

3.2 Related work

In this section, we present a brief review of different tweet expansion and hashtag recommendation approaches. Earlier studies on tweet expansion may be grouped

broadly into two; (i) expansion using text drawn from external sources, and (ii) pooling related tweets together by common hashtags, mentions or interactions of tweets.

3.2.1 Tweet Expansion using Text from External Sources

Contents from external sources such as news articles, Wikipedia, and URLs present in tweets are used to semantically enrich short and noisy tweets, which is then used for several tasks such as user profiling for news recommendation [85, 110, 111], event detection [112, 113] and named entity recognition [114, 115]. Abel et al. [85] have expanded tweets with the news articles using URL-based and content-based strategies to enrich the semantics of tweet by using the entities, topics, and events present in the corresponding news articles. In content-based strategy, similarity between a news article and a tweet pair is measured by TF-IDF score over bag-of-words representation using different features such as content of the tweet, content of news article, hashtags of the tweet and named entities of the news article. The expanded semantics of tweets with news articles helps in construction of more meaningful user profile as compared to using semantics of tweets only. Further, Lu et al. [110] used Explicit Semantic Analysis *(ESA)* [116] method to compute semantic similarity between a tweet and Wikipedia concepts and then represent each tweet as a weighted vector of Wikipedia concepts. It has been observed that tweet representation obtained using ESA enhances user profiling and tweet recommendation performance. Similarly, Kang et al. [111] have also exploited ESA and Wikipedia concept structure to model users' interest over different topics of news publication.

Guo et al. [33] expanded tweets with contexts extracted from news articles using Weighted Textual Matrix Factorization Graph *(WTMF-G)* over the features such as hashtags of the tweets, entities in the summary of news and time of publication. WTMF-G has improved performance of finding related news article for a tweet compared to other baseline models such as Information Retrieval (IR) based models, LDA based models, and a variant of WTMF-G. Romero et al. [112] extracted the *named entities* and *locations* using Open Calais [2], frequent keywords, and representative keywords using TF and TF-IDF as important event descriptive features and extracted the knowledge about these features from Linked Open Data cloud (DBPedia, YAGO). The short text with textual features and semantically enriched features are then passed to Naive Bayes and SVM classifier for event classification.

[2] http://www.opencalais.com

After enrichment of location and domain representative terms, the paper reported an improved classification performance for planned events like Football World Cup 2010, 2012 Olympic Games, etc. Morabia et al. [113] proposed SEDTWik to detect events from tweets using Wikipedia titles in four steps: i) extraction of meaningful tweets and hashtags segment ii) detection of bursty segment iii) clustering bursty segments into events and iv) event summarization. SEDTWik tokenized tweets and hashtags into informative segments using Wikipedia's title. Thereafter, bursty segments are extracted for each time window using different features such as the count of the segments, number of diverse users tweeting the segment, number of re-tweets of the segment and the number of followers for the user tweeting the segment. The bursty segments are then clustered into events using a variation of shared nearest neighbor algorithm [117]. Further, non-news worthy events cluster are pruned with the help of Wikipedia. Lastly, event clusters are summarized using LexRank algorithm [118]. SEDTWik reported an improvement in the performance of event detection in terms of precision in comparison to state-of-the-art methods. Gattani et al. [114] proposed Doctagger to perform following three tasks on a tweet: a) extraction of named entities, b) linking of extracted named entities to real time knowledge-base (Wikipedia, Yahoo! Stocks, Adam for health, MusicBrainz for music albums, etc.), and c) classification and tagging of the tweet. After enriching semantics of tweets (context, social signal, and handcrafted rules), Doctagger improved the performance of named entity recognition compared to Standford Named Entity Tagger [3] and entity linking in comparison with Open Calais [4]. Web context for a tweet is extracted by excerpting the title and first few lines of web pages mentioned in tweets. Similarly, contexts for a specific user at the time t is extracted as union of tags from his last k tweets. And, context of a hashtag h at the time t is extracted as union of tags associated to the tweets till time t mentioning hashtag h. Li et al. [115] proposed a framework named HybridSeg to improve the named entity recognition task in a tweet by segmenting it into meaningful chunk using stickiness score of each chunk. The stickiness score of each segment is calculated based on the probability of N-gram words in English corpus (Microsoft N-gram and Wikipedia corpus) or words in tweets batch (posted in short time duration). Expanding tweets with external sources can also be used to improve topic modeling performance. While expanding tweets with text/concepts from external sources, there is a need for effective methodologies to extract relevant text/concepts from external sources. Considering the heterogeneous nature of content in tweets such as misspelling, short form, long form, multilingual,

[3] https://nlp.stanford.edu/software/CRF-NER.html
[4] http://www.opencalais.com

code-mixed etc., devising an effective method to extract relevant text/concepts from external sources is a challenging task.

3.2.2 Tweet Expansion by Pooling Related Tweets

Different pooling approaches for enhanced topic modeling have been studied in the past. Considering the approaches adopted by the authors, we classify the studies as follows. (i) **Hashtag pooling**: it combines several tweets which share common hashtags into a single pseudo document. Mehrotra et al. [35], and Steinskog et al. [36] apply hashtag pooling over a collection of tweets and perform topic modeling. For tweets without any hashtags, Mehrotra et al. [35] labeled tweets using hashtags of similar tweets and perform pooling. Steinskog et al. [36] considered only the tweets with single hashtag in their study. (ii) **User-centric temporal pooling**: it assumes that a user is likely to post tweets on the same topic in a given day. Alp et al. in [37] pooled users' tweets and date pair and apply LDA. It is reported that LDA on user-date pair pooling tweets outperforms the no-pooling, user pooling and hashtag-based pooling tweets. (iii) **Pooling by interactions**: in this approach, a tweet and its reply tweets are pooled together. Alvarez-Melis et al. [38] have reported that pooling by interaction provides better topic modeling and retrieval performance. Ollagnier et al. [119] have pooled interaction tweets (tweet of the users mentioned in the replies), and combined it with the original tweet. It has been reported that adding reply tweets improves the topic modeling performance as compared to no-pooling, user pooling, hashtag pooling, and interaction pooling. (iv) **Other Pooling Approach:** in Hajjem et al. [120], if an Information Retrieval *(IR)* system detects overlapping of top-ranked search results for different query tweets, then query tweets are pooled to a single document. Authors reported improved F-Measure, NMI, purity score as compared to un-pooled tweet and pooled ones by hashtags.

While pooling-based tweet expansion approaches have shown improved topic modeling performance in compared to unexpanded tweet, it may result in topic drift as the texts in the pooled tweets may dominate the content of the original tweet. Moreover, it may also increase the dataset by many folds, as the same tweets are pooled many times in different tweet pools. In this study, as the objective is to expand the tweets with relevant hashtags only, we consider only the hashtag-based pooling.

3.2.3 Hashtag Recommendation

Different hashtag recommendation approaches on tweets have been adopted by authors in the past, which can be broadly grouped as below. a)**Using feature-based approach**: Based on textual content similarity, Zangerle et al. [121] and Li et al. [122] recommended hashtags for the target tweets from similar tweets. In Zangerle et al. [121], the similarity is calculated using cosine score over TF-IDF score while in Li et al. [122], the similarity is calculated using cosine score over word similarity matrix based on WordNet.

b) **LDA based generative models**: Krestel et al. [123] recommended tags for a document using LDA. The paper reported improved performance of LDA based method in predicting tags over rule-based method using bookmarking datasets crawled from Delicious [5]. Using a translational-based model, Liu et al. [124], recommended tags for a document, in which the content and its tags are modelled as a translation of each other. The paper reported results of tag recommendation over two corpora, namely BOOK corpus(containing book description and its tags) and BIBTEX corpus (containing papers description and its associated tags) to model the tags. Ding et al. [125] proposed a Topic-Specific Translation Model *(TSTM)* extending LDA model to use topic information of a tweet for aligning a word and hashtag. TSTM shows an improvement in predicting top-1 hashtag in terms of Precision, Recall, F-Measure as compared to Naive Bayes, LDA [123], IBM-1 [124]. Ding et al. [126] proposed Topic Translation Model *(TTM)* extending Twitter-LDA [32] to predict hashtags of a tweet using a translational model with an assumption that contents and hashtags both are talking about the same topic written in different languages. TTM sampled a hashtag using the topic associated with the tweet and current word for every word of a tweet. The experimental result shows improved performance in terms of precision, recall and F-Measure as compared to Naive Bayes, LDA based model [123], transnational-based model IBM1 [124] and TopicWA [125].

c) **Using deep learning-based models**: Tomar et al. [127] proposed a deep Feed Forward Neural Network *(FFNN)* using the distributed representation of a word in a tweet and further exploited to recommend hashtags for a tweet. Li et al. [128] used an LSTM layer to get the tweet representation from the sentence representation obtained using CNN in a tweet. This tweet representation is used to predict hashtags for a given tweet. The proposed model achieves improved Accuracy and Hitrate [129] compared to three layer FFN with ReLU(a variant of [127]), LSTM

[5] http://delicious.com

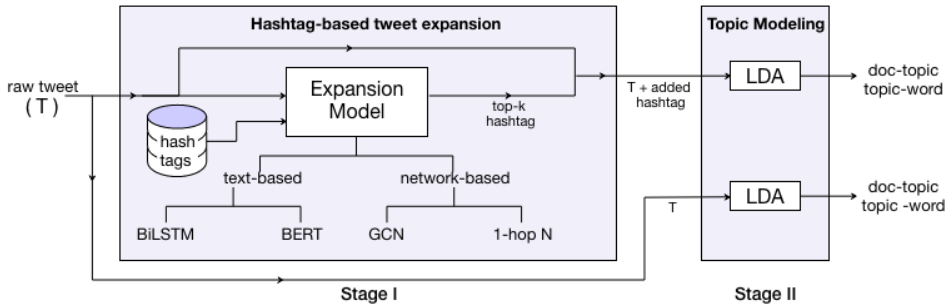


FIGURE 3.1: Framework diagram of Topic modeling over expanded tweets.

at the word level, RNN and GRU. Gong et al. [130] proposed a CNN-based model using a local and a global channel to incorporate the attention to topic-specific words. The local channel is used to get representation of topic trigger words, and a global channel is used to get representation of all the words. The tweet vector obtained by concatenation of local channel and global channel is used to predict hashtags for a given tweet. The proposed model shows improved performance for hashtag prediction in terms of Precision, Recall, F-Measure as compared to Naive Bayes model, LDA based model [123], translation-based model IBM-1 [124], TopicWA [125], TTM [126], only using CNN, and only using attention.

Motivated from the above, we have applied widely used sequence to sequence learning models, namely BiLSTM [108] and BERT [107] to predict semantically related hashtags in text-based approach. Further, we also used 1-hop nearest neighbor and GCN [108] based approach to predict semantically related hashtags.

3.3 Methodology

This section discusses the different approaches used for hashtag-based tweet expansion to improve topic modeling performance. Figure 3.1 presents the overall framework diagram for proposed hashtag-based tweet expansion approaches. At first, raw tweet (T) is passed though hashtag-based tweet expansion module (Stage-I) to extract top-k semantically related hashtags. The expansion module in the Stage-I in based on two different approaches, namely – a) text-based sequential models and b) network-based models to utilize the textual and structural properties of tweet in predicting semantically related hashtags to a given tweet (T). Further, The top-k hashtags are added to the raw tweet (T) and passed to the Topic Modeling module in Stage II. The objective of this setup is to study the

effect of different hashtags-based tweet expansion approaches in topic modeling performance in contrast to raw tweets.

In the subsequent subsection, text-based sequential models(BiLSTM) [105] and Bidirectional Encoder Representations from Transformers (BERT) [107]) and network-based models (1-hop nearest neighbor and Graph Convolution Network *(GCN)* [108]) used in Stage-I for tweet expansion approaches using related hashtags are discussed. The brief details about a popular topic modeling method namely LDA, used in Stage-II of the framework diagram, is already discussed in section 2.2 of the thesis.

3.3.1 Text-based Sequential models for Tweet expansion

This subsection discusses the process of predicting semantically related hashtags of a tweet using text-based sequential models, namely Bidirectional Long Short-Term Memory *(BiLSTM)* [105] and Bidirectional Encoder Representations from Transformers *(BERT)* [107]. Given an input tweet with words sequence w_1, w_2, ..., w_k, we associate each word with an embedding representation $\mathbf{x}_t \in \mathbb{R}^d$ vector where $t \in [1, k]$ and d is the word embedding dimension. The tweet can then be represented as $\mathbf{X} \in \mathbb{R}^{k \times d}$ where the t^{th} index of $\mathbf{X}$ is the x_t vector. The word embedding vectors are passed to sequence learning models such as BiLSTM and BERT to represent the input tweet. The BiLSTM model encode the representation of the word sequence by concatenating the outputs of two LSTM, namely LSTM-forward and LSTM-backward. Each LSTM model consists of a repeating unit called memory cell, which takes current word, previous hidden state, previous cell state ($\mathbf{x}_t$, $\mathbf{h}_{t-1}$, $\mathbf{c}_{t-1}$) as input and produces current hidden state, cell state information i.e. $(\mathbf{h}_t, \mathbf{c}_t) = LSTM(\mathbf{x}_t, \mathbf{h}_{t-1}, \mathbf{c}_{t-1})$. The memory cell of LSTM consists of three gates, namely forget gate (f_t), input gate (i_t), and output gate (o_t). The transition equation of LSTM memory cell can be represented as follows:

$$\begin{aligned}
\mathbf{f}_t &= \sigma(W_f \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_f) \\
\mathbf{i}_t &= \sigma(W_i \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_i) \\
\tilde{\mathbf{c}}_\mathbf{t} &= tanh(W_c \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_c) \\
\mathbf{c}_t &= \mathbf{f}_t * \mathbf{c}_{t-1} + \mathbf{i}_t * \tilde{\mathbf{c}}_t \\
\mathbf{o}_t &= \sigma(W_o \cdot [\mathbf{h}_{t-1}, \mathbf{x}_t] + \mathbf{b}_o) \\
\mathbf{h}_t &= \mathbf{o}_t * tanh(\mathbf{c}_t)
\end{aligned}$$

TABLE 3.1: Hyperparameters for sequence learning methods.

Sequence learning models	**Hyper-parameter**
Bidirectional Long Short Term Memory (BiLSTM)	64 LSTM Units, *ReLU* Activation Function, 40 epochs, `k` = 30 words, `d` = 64
Bidirectional Encoder Representations from Transformers (BERT)	64 hidden size, *GeLU* Activation Function, 8 multi-head attentions, 40 epochs, `k` = 40 words, `d` = 64

where W_f, W_i, W_c, W_o are weight matrices, b_f, b_i, b_c, b_o are bias vectors, and σ (*Sigmoid*), and *tanh* are the activation functions. The LSTM-forward model ($LSTM^{(f)}$) process the word sequence from left to right w_1, w_2, ..., w_k, whereas LSTM-backward ($LSTM^{(b)}$) process the word sequence in from right to left, i.e. w_k, w_{k-1}, ..., w_1. For each time step t, the transition equation of LSTM-forward and LSTM-backward is as follows:

$$(\mathbf{h}_t^{(f)}, \mathbf{c}_t^{(f)}) = LSTM^{(f)}(\mathbf{x}_t, \mathbf{h}_{t-1}^{(f)}, \mathbf{c}_{t-1}^{(f)})$$
$$(\mathbf{h}_t^{(b)}, \mathbf{c}_t^{(b)}) = LSTM^{(b)}(\mathbf{x}_t, \mathbf{h}_{t+1}^{(b)}, \mathbf{c}_{t+1}^{(b)})$$

BiLSTM concatenate hidden state obtained by LSTM-forward (($h_t^{(f)}$) and LSTM-backward ($h_t^{(b)}$) to produce the representation of word w_t, i.e., $\mathbf{h}_t = (\mathbf{h}_t^{(f)} \oplus \mathbf{h}_t^{(b)})$. The representation produced by the BiLSTM at t=k encodes the completes sequence and can be written as $\mathbf{h}_k = (\mathbf{h}_k^{(f)} \oplus \mathbf{h}_k^{(b)})$.

Along with the BiLSTM, this study also considers using Bidirectional Encoder Representations from Transformers, more commonly known as BERT [107], to generate the text representation of an input tweet T. Given the word representation of the tweet $\mathbf{X} \in \mathbb{R}^{k \times d}$ and positional encoding of words position in the tweet $\mathbf{P} \in \mathbb{R}^{k \times d}$ calculated using Equation 3.1, the BERT model transforms it to $\mathbf{Z} \in \mathbb{R}^{k \times d}$ a representation incorporating the bidirectional semantic information of the word sequences.

$$\mathbf{P}_{pos,i} = PE(pos, i) \quad i \in (1, d),\ pos \in (1, k)$$
$$PE(pos, i) = \begin{cases} sin(pos/10000^{2i/d}) & \texttt{if} i \texttt{ is even} \\ cos(pos/10000^{2i/d}) & \texttt{otherwise} \end{cases} \tag{3.1}$$

Each transformer block in BERT with mh multi-head attentions at a time $t \in (0, l)$ can be defined as:

$$\forall i \in (1, mh) \begin{cases} \mathbf{Q}_i = \mathbf{W}_{qi} \cdot \mathbf{Z}_t \\ \mathbf{K}_i = \mathbf{W}_{ki} \cdot \mathbf{Z}_t \\ \mathbf{V}_i = \mathbf{W}_{vi} \cdot \mathbf{Z}_t \\ \mathbf{Y}_i = softmax(\frac{\mathbf{Q}_i \cdot \mathbf{K}_i^T}{\sqrt{d}})\mathbf{V}_i \end{cases} \tag{3.2}$$
$$\mathbf{Y} = \mathbf{Y}_{1:mh} \quad \mathbf{Y} \in \mathbb{R}^{mh \cdot k \times d}$$
$$\mathbf{Z}_{t+1} = \mathbf{W}_{zt} \cdot \mathbf{Y}$$

where $\mathbf{Z}_0 = \mathbf{X} + \mathbf{P}$, $\{\mathbf{W}_{qi}, \mathbf{W}_{ki}, \mathbf{W}_{vi}\} \in \mathbb{R}^{k \times k}$ and $\mathbf{W}_{zt} \in \mathbb{R}^{k \times k \cdot mh}$ are the weighted parameters of the transformer block, and $\oplus$ denotes the concatenation operator. The output of the last transformer block $\mathbf{Z}_l$ is taken as the final output of the BERT model. To represent in the vector space, $\mathbf{Z}_l$ is being reshaped into $\mathbf{z}_{seq} \in \mathbb{R}^{k \cdot d \times 1}$ vector as an input to next layer. The whole operation can be represented as:

$$\mathbf{z}_{bert} = BERT(\mathbf{Z}_0, \theta) \tag{3.3}$$

where θ represents the hyperparameters such as l number of encoders, mh number of multi-head attentions, d hidden layer dimensions. We have considered $l = 8$ transformer blocks and $mh = 8$ multi-head attentions as used in default BERT setup. $\mathbf{z}_{bert}$ represent the encoded representation of the input tweet T.

The encoded representation obtained by BiLSTM and BERT capture the semantic and syntactic relations of the word sequences in the tweet. The encoded output is then used to predict related hashtags given the input text sequence through a Feed Forward Neural *FFN)* network and a Softmax activation function. The hyperparameters for training sequence learning models (i.e., BiLSTM and BERT) are provided in Table 3.1. We use Keras[6] and Transformer[7] Python libraries to build BiLSTM and BERT. We use Categorical Cross-Entropy loss function and Adam optimizer to train the sequence model.

To train the sequence learning model, we considered tweets having at least one hashtag. From this collection, we curate the training dataset by omitting the hashtags present in the tweet and set them as the target hashtags for prediction. For example, a tweet "*@firstpost Not a single proof gvn. #Pakistan is asking*

[6] https://keras.io/
[7] https://huggingface.co/transformers/

TABLE 3.2: Bi-LSTM training dataset example. Original tweet: @firstpost Not a single proof gvn. #Pakistan is asking for international inquiry of #UriAttack but #Modi Govt refusing. Weird. @UN #TrumpWon.

S.No	Input text sequence	Target hashtag
1	@firstpost Not a single proof gvn . is asking for international inquiry of #UriAttack but #Modi Govt refusing . Weird . @UN #TrumpWon	#Pakistan
2	@firstpost Not a single proof gvn . #Pakistan is asking for international inquiry of but #Modi Govt refusing . Weird . @UN #TrumpWon	#UriAttack
3	@firstpost Not a single proof gvn . #Pakistan is asking for international inquiry of #UriAttack but Govt refusing . Weird . @UN #TrumpWon	#Modi
4	@firstpost Not a single proof gvn . #Pakistan is asking for international inquiry of #UriAttack but #Modi Govt refusing . Weird . @UN	#TrumpWon

for international inquiry of #UriAttack but #Modi Govt refusing. Weird. @UN #TrumpWon", we get four instances of training set (input text sequence and target hashtag) as shown in Table 3.2. The process of hashtag prediction can be mathematically represented as:

$$\mathbf{v} = Sequence\_model(\mathbf{D})$$
$$\mathbf{H}_{score} = Softmax(FFN(\mathbf{v}))$$

where *Sequence_model* is any sequence learning model (we have experimented using BiLSTM and BERT), $\mathbf{D}$ is the document matrix of `k` x `d` size composed of the input sentence of `k` words with its embedding `d` dimensions, FFN is feed forward neural network layer, and $Softmax$ is the neural activation function. For input sentences having `w` < `k` words, we padded (`k` - `w`) numbers of $\mathbf{0}$ vectors of `d` dimensions. The *Sequence_model* function transforms the $\mathbf{D}$ matrix to $\mathbf{v}$ vector. Finally, we got the $\mathbf{H_{score}}$ vector of `h` classes to predict the scores of the target hashtags. After training the sequence learning model, we expand the original tweets by predicting semantically related hashtags and select the top `n` hashtags using the $\mathbf{H_{score}}$. The trained sequence model can also predict semantically related hashtags for tweets that do not have any hashtags. Finally, the selected `n` hashtags are added to the original tweet for topic modeling using LDA.

3.3.2 Network based graphical models for Tweet expansion

This subsection discusses the process of predicting semantically related hashtags of a tweet using a network-based graphical approach. A tweet can be represented in a graph structure using co-occurrence relations of words in the tweet. Let $G(V, E)$ represent an undirected weighted co-occurrence network obtained from a tweet, where $V = \{w_1, w_2, \ldots, w_i, \ldots, w_N\}$ are unique words in the tweet, N is the total number of unique words in a tweet, $E = \{(w_i, w_j)$: $w_i, w_j \in$ V, w_i, w_j co-occur in the tweet$\}$, and $A_{(N*N)}$ is an adjacency matrix where $A[w_i, w_j]$ represents number of times word w_i and w_j co-occur in a tweet. We used superscript to differentiate graph of individual tweets, for example $G(V, E)^{(t)}$, $A^{(t)}$ represents the tweet graph and adjacency matrix of a0 tweet t. We combine all the tweet graphs to form a global network $\mathbf{G}(\mathbf{V}, \mathbf{E})$, where $\mathbf{V} = V^{(1)} \cup V^{(2)} \cup \cdots \cup V^{(T)}$, $\mathbf{E} = E^{(1)} \cup E^{(2)} \cup \cdots \cup E^{(T)}$, T is the total number of tweets in the corpus and adjacency matrix as follows:

$$\mathbf{A}[w_i, w_j] = \sum_{t=1}^{t=T} \begin{cases} A^{(t)}[w_i, w_j] & \text{if } w_i \text{ and } w_j \in t \\ 0 & \text{Otherwise} \end{cases}$$

where $w_i, w_j \in \mathbf{V}$. To discover semantically related hashtags for a given tweet network, this study explores two graph-based methods, namely 1-hop nearest neighbor and Graph Convolution Network *(GCN)* [108]. The graph $\mathbf{G}$ is considered for retrieving semantically related hashtags using 1-hop nearest neighbor-based method. While the tweet graphs $G^{(t)}$ $(t \in [1, T])$ are considered for retrieving semantically related hashtags using GCN based method.

3.3.2.1 1-hop Nearest Neighbor-based Tweet expansion

This method finds related hashtags of a tweet by calculating the weighted score between all words in the tweet against all target hashtags H of the above tweet network $\mathbf{G}$ using weighted adjacency matrix $\mathbf{A}$. An entry $\mathbf{A}[w_i, w_j]$ represents the number of co-occurrence of words w_i and w_j in the tweet corpus. For a tweet t, we score each target hashtag h by summing the weights of the adjacency matrix $\mathbf{A}$ as $score(t, h) = \sum_{word \in t} \mathbf{A}(word, h)$, $h \in H$, where H is the set of unique hashtags in the graph $\mathbf{G}$. This scoring function captures the 1-hop neighbor distance of a tweet t with respect to a hashtag h. The intuition of this scoring function is to measure the semantic relationship of target hashtags with the given tweet t. We

expand each tweet with top k semantically related hashtags in H using the above scoring function.

3.3.2.2 Graph Convolution Network-based Tweet expansion

The 1-hop based nearest neighbor tweet expansion method exploits explicit relations of words present in the tweet network **G** to find semantically related hashtags. Therefore, the above method only considers the semantics based on word co-occurrence relations. To capture the implicit relations between words in a graph, recent trends employ a graph neural network-based paradigm, which represent words in latent space. Zhang et al. [108] have used multilayer Graph Convolution Networks *(GCN)* [109] for graph classification task. The GCN model proposed by Kipf and Welling [109] works on a single graph structure (**G**) that captures the local semantics of the nodes. However, Zhang et al. [108] method is able to represent graphs of arbitrary structures ($G^{(t)}$ $(t \in [1, T])$). To capture the global semantics of the node in different graph instances (tweets), they proposed an algorithm named *SortPooling* similar to Weisfeiler-Lehman node coloring algorithm [131] for sorting vertex features that capture the global node information. This study considers the tweet representation generated using GCN based method proposed by Zhang et al., which is mathematically described below.

Given a graph G, the encoder transforms the input graph to a stochastic matrix Z of $k \times m$ dimensions, where k is the number of nodes and m is the number of output units in GCN. GCN takes two input matrices A and X, where A is the adjacency matrix of the network, and X is a feature matrix (word embedding) of dimension $k \times d$. GCN can be mathematically represented as follows:

$$\hat{X} = GCN(X, A) = \sigma(\tilde{A}XW)$$
$$\tilde{A} = D^{(-\frac{1}{2})}AD^{(-\frac{1}{2})}$$

where $\tilde{A}$ is the symmetrically normalized adjacency matrix, D is the degree matrix, W is the weight parameter of the neural network, and σ is the activation function. In this study, we use ReLU activation function and employ a two-layer GCN defined as follows:

$$GCN(X, A) = \sigma(\tilde{A}\sigma(\tilde{A}XW_1)W_2)$$

where W_1 and W_2 are the weight parameters for the first and second layers of the GCN. The GCN embedding matrix Z is then generated using linear combination

of two GCNs sharing the weight of the first layer.

$$\mu = GCN_{\mu}(X, A) = \sigma(\tilde{A}\sigma(\tilde{A}XW_1)W_2)$$
$$\delta = GCN_{\delta}(X, A) = \sigma(\tilde{A}\sigma(\tilde{A}XW_1)W_3)$$

$$Z = \mu + \delta * \epsilon$$
$$Z_{Sort} = SortPool(Z)$$
$$\mathbf{g} = MaxPool({Z_{Sort}}^T)$$
$$\mathbf{H}_{score} = Softmax(FFN(\mathbf{g}))$$

where $\epsilon \sim N(0, 1)$. We perform SortPooling [108] over the Z matrix to sort the latent representation of the nodes, and then apply MaxPool over the transpose of Z_{Sort} the matrix to represent the input graph as $\mathbf{g}$ a vector. We, then, predict the target hashtags using Feed Forward Neural Network *(FFN)* with Softmax activation function over the graph representation $\mathbf{g}$.

3.4 Experimental Setups

This section presents the datasets characteristics in terms of number of classes, number of tweets with hashtags, and words overlapping between classes. Furthermore, it presents the different experimental setups along with details of hyperparameters used.

3.4.1 Dataset

For this study, we set up a tweet crawler using Tweepy streaming API [8] to collect tweets corresponding to daily trending hashtags, keywords from India and the tweets of popular and active Indian users. Thereafter, we filtered the tweets corresponding to major events happening in India using representative hashtags and keywords. We manually identified the representative hashtags and keywords for each event class using tweet co-occurrence matrix and observing the daily and hourly trends from Twitter and Trends24 website [9]. We have curated real-world event related tweet dataset from diverse topics such as Attacks (Uri Attack[10],

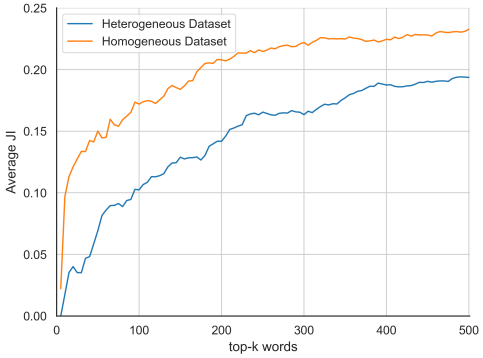
[8] Tweepy streaming API
[9] Trends24
[10] Uri Attack

TABLE 3.3: Homogeneous and Heterogeneous dataset description.

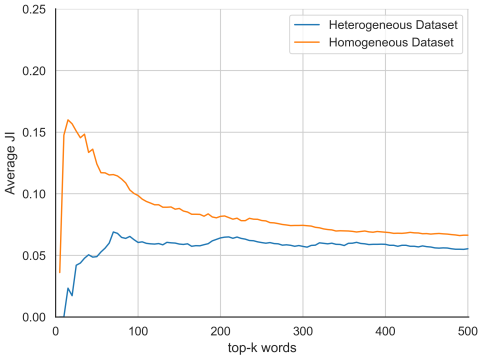
S.No	Class Name	# of tweets	# of tweets with hashtag
	Heterogeneous Dataset		
1	GSTN	22512	7135
2	Attack	19336	12098
3	CAB protest	18434	18434
4	BiharElection2020	15600	15600
	Homogeneous Dataset		
1	Surgical Strike	7585	7543
2	Kashmir Unrest	5947	2361
3	Pathankot Attack	5057	1458
4	Syria Crisis	1012	408
5	Uri Attack	747	736

Pathankot attack[11], Kashmir unrest[12], Syria Crisis[13], Surgical strike[14]), Government policy (Citizenship Amendment Bill (CAB)[15], Goods and Service Tax Network (GSTN)[16]) and Election (Bihar Elections2020[17]). The dataset generation strategy used in this study is similar to Task-4 (Sentiment Analysis in Twitter) of SemEval-2017 datasets [18]. Afterwards, tweets are assigned to an event class associated with the hashtags or keywords contained in it. Since Twitter does not restrict in choosing hashtag or keywords, spammers also tweet unrelated content with trending hashtags and keywords to seek attention. For example, user tweeted about #jallikattu protest[19] along with trending GSTN hashtags. Hence, it is important to filter noisy and spammed tweets before performing topic modeling. We apply n-grams based approach to filter noisy and spammed tweets. We mark a few of the frequently occurring n-grams as noisy with respect to the tagged topics. We then remove tweets containing the noisy tagged n-grams from respective topics. After removing spam tweets, we finally consider 77,990 tweets for our experimental studies.

[11] Pathankot attack
[12] Kashmir unrest
[13] Syria Crisis
[14] Surgical strike 2016 by Indian Army
[15] CAB 2019
[16] GSTN
[17] Bihar Elections2020
[18] Task-4 of SemEval-2017
[19] Jallikattu protest



(A) Average Jaccard Index *(JI)* similarity between all class pairs using all words (except user mentions).



(B) Average Jaccard Index *(JI)* similarity between all class pairs using only hashtags.

FIGURE 3.2: Average Jaccard Index *(JI)* similarity between all class pairs over Heterogeneous and Homogeneous Dataset using all words (except user mentions) and only hashtags.

Using the above-mentioned approach, we curated two types of datasets from the above tweet collections: a) `Heterogeneous dataset` – tweets collected from dissimilar topics, b) `Homogeneous dataset` – tweets collected from similar topics. Table 3.3 shows the statistics of Heterogeneous and Homogeneous tweet datasets. We performed pre-processing of the tweets such as conversion of words to lowercase, removal of URLs, punctuation, and emoticons from the tweet text using NLTK toolkit[20]. The Heterogeneous dataset has 75,882 tweets distributed under four classes, namely a) Goods and Services Tax Network *(GSTN)*, b) Attack: consisting of Uri Attack, Pathankot Attack, Surgical Strike, Kashmir Unrest, c) Citizenship Amendment Bill *(CAB)*, d) BiharElection2020. This dataset has 7,670 unique hashtags and 38,687 unique keywords after pre-processing. The Homogeneous dataset consists of 20,306 tweets distributed under five classes: Uri Attack, Pathankot Attack, Kashmir Unrest, Surgical Strike, and Syria Crisis. This dataset has 2,671 unique hashtags and 19,084 keywords after pre-processing. To study the characteristics of both the datasets, we incorporated Jaccard Index *(JI)* [132] similarity between classes to quantify the overlapping of keywords and hashtags in Homogeneous and Heterogeneous Datasets. Jaccard Index similarity between any two classes C_i and C_j considering `top-k` words for each of the classes can be defined as follows:

$$JI^k(C_i, C_j) = \frac{|S_i^k \cap S_j^k|}{|S_i^k \cup S_j^k|}. \tag{3.4}$$

[20] NLTK

where S_i^k and S_j^k represents the set of `top-k` words present in class label C_i and C_j respectively and k $\in \mathbb{N}$. Figure 3.2 presents the average JI score between all classes of Heterogeneous and Homogeneous datasets using all words (except user mentions) and only hashtags at different value of $k \in [1,500]$. Higher value of average JI indicates more overlapping of words and hashtags between classes. From the Figures 3.2a, 3.2b, we infer that the Homogeneous Dataset has more overlapping of words and hashtags compared to the Heterogeneous Dataset.

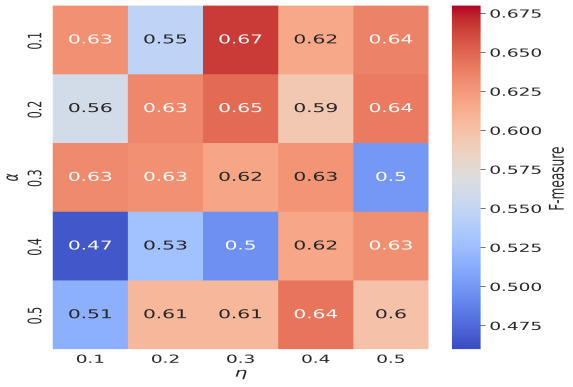


FIGURE 3.3: Comparison of LDA performance using F-measure at different values of α and η over Heterogeneous dataset.

3.4.2 Type of LDA setups

Given a tweet, the objective of the study is to identify topics of the tweet using LDA. We consider LDA over original unexpanded tweets as the baseline setup for performance comparison of LDA over our proposed tweet expansion methods. We consider five LDA setups corresponding to the input text type, namely:

- **Raw tweet (T)**: This setup takes the raw tweet after pre-processing as input to LDA model.
- **T+HashtagPool**: Tweets in hashtag pool corresponding to each hashtag of a raw tweet is added to the raw tweet.
- **T+BiLSTM**: The top `n` related hashtags predicted using BiLSTM model is added to the raw tweet.
- **T+BERT**: The top `n` related hashtags predicted using BERT model is added to the raw tweet.

- **T+1-hop N**: The top `n` semantically related hashtags selected using the 1-hop based nearest neighbor is added to the raw tweet.
- **T+GCN**: The top `n` semantically related hashtags using the Graph Convolution Network *(GCN)* based tweet embedding is added to the raw tweet.

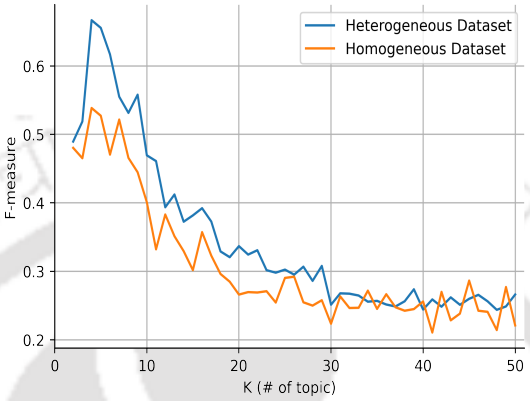


FIGURE 3.4: Comparison of LDA performance using F-measure at different topics (K) over Heterogeneous and Homogeneous Dataset.

For all the LDA setups mentioned above, we choose the Dirichlet hyperparameters α (document-topic distribution) as 0.1 and η (topic-word distribution) as 0.3. We set the above hyperparameters based on the empirical evaluation of the performance of LDA in the interval [0.1, 0.5] with a step of 0.1. Figure 3.3 presents the LDA performance over Heterogeneous dataset using different values of α and η. To choose the number of topics for the Heterogeneous and Homogeneous dataset, we empirically evaluated the performance of LDA at different values of number of topics (K). Figure 3.4 presents the comparative performance of LDA in terms of F-measure for K in in the interval of [2,50] with a step of 1. From the figure 3.4, we observe that maximum performance (in terms of F-measure) is obtained at topic 4 for both Heterogeneous and Homogeneous datasets, respectively. However, the F-measure value for Homogeneous dataset at topic 4 and 5 is very close. Therefore, we have set the number of topics equal to number of classes for Heterogeneous and Homogeneous datasets as 4 and 5 respectively, to measure the efficacy of our proposed topic modeling methods to represent the original class distribution. We use collapsed Gibbs sampling to estimate the parameters of LDA. We run upto 200 iterations to train the LDA model for all setups except for T+HashtagPool over the Heterogeneous dataset. For T+HashtagPool over Heterogeneous dataset, we ran LDA upto 24 iterations, as there was no significant change in the perplexity (topic modeling evaluation metric used in [2]) in successive iterations.

3.5 Results and Observations

In this section, we first discuss the effect of different word types on the performance of LDA. We further discuss the comparative results of our proposed tweet expansion methods over the baseline (raw tweet).

3.5.1 Effect of hashtags on LDA performance over tweets

In this subsection, we study the influence of hashtags and user mentions[21] by performing the following analysis; i) perform LDA over raw tweets, ii) remove hashtags from the raw tweets and perform LDA (WT-H), iii) remove user mentions from the raw tweet and performs LDA (WT-M), and iv) remove hashtags and user mentions from the raw tweet (WT-H-M) and perform LDA. Table 3.4 shows the comparison of LDA performance using different setups of raw tweets, WT-H, WT-M, and WT-H-M over Heterogeneous and Homogeneous datasets. In case of Heterogeneous dataset, performance after removing hashtags is decreased by 4%, 16%, 2% and 5% in terms of F-Measure, Rand Index, Normalized Mutual Information *(NMI)* and Jaccard Coefficient *(JC)* respectively. The performance after removing user mentions is increased by 22%, 38%, 7% and 32% respectively in terms of F-Measure, NMI, Rand Index, and JC respectively, indicating this to be the most noisy features. The performance after removing both hashtags and mentions is increased by 6%, 4%, 1% and 8% respectively. Therefore, hashtags are the most informative features in case of Heterogeneous dataset, as the LDA performance drop is maximum after removing hashtags. However, mentions in case of Heterogeneous dataset are noisy features, as there is improvement in LDA performance after removal of user mentions.

In case of Homogeneous dataset, the performance after removing hashtags is decreased by 5%, 19%, 3% and 6% respectively in terms of F-Measure, NMI, Rand Index and JC respectively. The performance after removing user mentions is decreased by 5%, 7%, 2%, and 7% in terms of F-Measure, NMI, Rand Index, and JC respectively. The performance after removing both hashtags and user mentions is decreased by 8%, 23%, 3%, and 10% in terms of F-Measure, NMI, Rand Index and JC respectively. In case of Homogeneous dataset too, hashtags are important feature for determining topics as there is significant decrease in LDA performance after removing hashtags.

[21] Words starting with @

TABLE 3.4: Effect of different entities combination using LDA performance in tweet over Heterogeneous and Homogeneous dataset.

Dataset_name	Setup	F-Measure (%)	NMI(%)	Rand Index (%)	JC(%)
Heterogeneous dataset	raw tweet	49.36	34.64	73.65	32.77
	WT-H	**47.38% (-4%)**	**29.23 (-16%)**	**72.44 (-2%)**	**31.05 (-5%)**
	WT-M	60.25 (+22%)	47.93 (+38%)	78.63 (+7%)	43.11 (+32%)
	WT-H-M	52.24 (+6%)	35.93 (+4%)	74.44 (+1%)	35.36 (+8%)
Homogeneous dataset	raw tweet	52.08	40.08	76.06	35.21
	WT-H	**49.64 (-5%)**	**32.61 (-19%)**	**74.14 (-3%)**	**33.01 (-6%)**
	WT-M	49.23 (-5%)	37.186 (-7%)	74.78 (-2%)	32.66 (-7%)
	WT-H-M	48.12 (-8%)	30.70 (-23%)	73.54 (-3%)	31.69 (-10%)

3.5.2 Effect of different tweet expansion approaches tweets in LDA performance

This subsection presents a comparative study of LDA performance using different tweet expansion approaches over raw tweets (Homogeneous and Heterogeneous datasets). The different approaches of tweet expansion shown in the Tables 3.5 and 3.6 are BiLSTM based (T+BiLSTM), 1-hop nearest neighbor based (T+1-hop N), GCN based (T+GCN), and BERT based (T+BERT). We conduct the experiment with different value of added hashtags (top `n`) in set of {2, 4, 6, 8, 10} to study its impact in LDA performance in terms of F-Measure, Normalized Mutual Information *(NMI)*, Rand Index and Jaccard Coefficient (JC). The brief description about the different evaluation metrics (F-measure, Rand Index, JC and NMI) is presented in the section 2.3 of the Chapter-2. We also compare the performance of LDA over expanded tweets from hashtag pooling (T+HashtagPool), and other tweets expansion approaches.

In case of Heterogeneous dataset (as given in Table 3.5) LDA performance on raw tweets in terms of F-Measure, NMI, Rand Index, and JC are 66.67%, 54.72%, 82.49% and 50% respectively. Using T+BiLSTM approach, all the setups except Added-2 hashtags perform better than raw tweet in terms of F-Measure, NMI, Rand Index, and JC. The performance of T+BiLSTM increases as we increase the number of hashtags added and reach at it's maximum for Added-8 hashtags with an improvement of 19%, 35%, 8%, and 32% over raw tweets in terms of F-Measure, Rand Index, NMI and JC respectively. T+BiLSTM reaches its maximum performance at Added-8 hashtags, thereafter, it decreases for Added-10 hashtags. Similarly, for T+GCN approach, all the setups except Added-2 hashtags perform better than the raw tweets. The performance increases as we increase

TABLE 3.5: Comparative results of LDA over raw tweet (T) and proposed different approaches of expanded tweet over Heterogeneous dataset in terms of F-Measure, Rand Index, NMI, JC and TC.

Types of methods	# hashtags added	F-Measure (%)	NMI (%)	Rand Index (%)	JC (%)	TC
Raw tweet (T)	–	66.67	54.72	82.49	50.00	0.2403
T+Hashtag Pooling	–	64.52 (-3%)	57.87 (+6%)	81.19 (-2%)	47.62 (-5%)	-0.3152(-231%)
T+BiLSTM	Added-2	64.07 (-4%)	53.79 (-2%)	80.58 (-2%)	47.14 (-6%)	+0.2041(-15%)
	Added-4	75.32 (+13%)	68.08 (+24%)	87.32 (+6%)	60.41 (+21%)	0.1544(-36%)
	Added-6	77.81 (+17%)	71.65 (+31%)	88.59 (+7%)	63.68 (+27%)	0.1504(-37%)
	Added-8	**79.5 (+19%)**	**73.94 (+35%)**	**89.5 (+8%)**	**65.97 (+32%)**	0.1695(-30%)
	Added-10	66.85 (0%)	63.49 (+16%)	81.98 (-1%)	50.21 (+6%)	0.1409(-41%)
T+1-hop N	Added-2	70.47 (+6%)	61.96 (+13%)	84.59 (+3%)	54.41 (+9%)	0.2087(-13%)
	Added-4	75.18 (+13%)	66.43 (+21%)	87.22 (+6%)	60.23 (+20%)	0.0671(-72%)
	Added-6	76.71 (+15%)	69.67 (+27%)	88.05 (+7%)	62.23 (+24 %)	0.0389(-84%)
	Added-8	79.99 (+20%)	71.28 (+30%)	89.8 (+9%)	66.66 (+33%)	0.0368(-85%)
	Added-10	**82.65 (+24 %)**	**75.20 (+37%)**	**91.2 (+11%)**	**70.43 (+41%)**	**0.469 (-80%)**
T+GCN	Added-2	64.46 (-3%)	60.02 (+10%)	79.95 (-3%)	47.56 (-5%)	0.1861(-23%)
	Added-4	87.31 (+31%)	80.43 (+47%)	93.4 (+13%)	77.48 (+55%)	0.1566(-34%)
	Added-6	**89.53 (+34%)**	**82.76 (+51%)**	**94.59 (+15%)**	**81.04 (+62%)**	0.1146(-52%)
	Added-8	85.15 (+28%)	77.54 (+42%)	92.36 (+12%)	74.14 (48%)	0.0708(-71%)
	Added-10	81.96 (+23%)	73.82 (+35%)	90.74 (+10%)	69.44 (39%)	0.0635(-74%)
T+ BERT	Added-2	**63.48 (-5%)**	**51.11 (-7%)**	**80.51 (-2%)**	**46.5 (-7%)**	0.1407(-41%)
	Added-4	59.51 (-11%)	46.61 (-15%)	78.14 (-5%)	42.36 (-15%)	-0.0217(-109%)
	Added-6	58.7 (-12 %)	44.27 (-19%)	77.9 (-6%)	41.54 (-17%)	-0.0574(-124%)
	Added-8	50.38 (-24%)	35.17 (-36%)	73.79 (-11%)	33.67 (-33%)	-0.0559(-123%)
	Added-10	56.35 (-15%)	42.56 (-22%)	76.4 (-7%)	39.23 (-22%)	-0.0454(-119%)

the number of hashtags added and achieves its maximum for Added-6 hashtags with an improvement of 34%, 51%, 15%, and 62% over raw tweets in terms of F-Measure, Rand Index, NMI and JC respectively. After reaching its maximum performance at Added-6 hashtags, it starts decreasing for Added-8 hashtags and Added-10 hashtags. One of the possible reason for lesser performance for Added-2 hashtags of T+BiLSTM and T+GCN as compared to raw tweet, is that both T+BiLSTM and T+GCN is biased towards predicting hashtags that is already contained in the tweet, hence adding no extra information. And for large value of `top-n`, the probability of hashtag prediction score is very low, causing a decrease in performance for Added-10 hashtags for T+BiLSTM and Added-8 and Added-10 hashtags for T+GCN. For T+1-hop N, all the setups performs better than the raw tweet. The performance continues to increase as we increase the number of hashtags added and reach at its maximum for Added-10 hashtags with an improvement of 24%, 37%, 11%, and 41% over raw tweets in terms of F-Measure, Rand Index, NMI and JC respectively. For T+BERT approach, all the setups have inferior performance than raw tweets in terms of F-Measure, NMI, Rand Index, and JC. The best performance of BERT approach is at Added-2 hashtags with a decrease in performance by 5%, 7%, 2%, 7% in terms of F-Measure, NMI, Rand Index and JC respectively. Possible reasons behind inferior performance of BERT-based approach may be smaller datasets size and less number of training examples for

TABLE 3.6: Comparative results of LDA over raw tweet (T) and proposed different approaches of expanded tweet over Homogeneous dataset in terms of F-Measure, Rand Index, NMI, JC and TC.

Types of methods	# hashtags added	F-Measure (%)	NMI (%)	Rand Index (%)	JC (%)	TC
Raw tweet (T)	–	52.72	38.20	76.75	35.8	0.1003
T+Hashtag Pooling	–	**59.31 (+13%)**	**42.72 (+12%)**	**78.28 (+2%)**	**42.15 (+18%**	-0.1796(+79%)
T+BiLSTM	Added-2	**67.09 (+27%)**	**52.57 (+38%)**	**82.56 (+8%)**	**50.48 (+41%)**	0.1574(+57%)
	Added-4	54.14 (+3%)	40.83 (+7%)	76.64 (0%)	37.12 (+4%)	0.1465(+46%)
	Added-6	56.45 (+7%)	43.13 (+13%)	78.11 (+2%)	39.33 (+10%)	0.1523(+52%)
	Added-8	49.14 (-7%)	36.47 (-5%)	74.6 (-3%)	32.57 (-9%)	0.1196(+19%)
	Added-10	53.04 (+1%)	38.58 (+1%)	76.53 (0%)	36.09 (+1%)	0.1273(+27%)
T+1-hop N	Added-2	48.47 (-8%)	34.75 (-9%)	73.42 (-4%)	31.98 (-11%)	0.0748(-25%)
	Added-4	60.28 (+14%)	47.63 (+25%)	79.84 (+4%)	43.14 (+21%)	0.0754(-25%)
	Added-6	66.23 (+26%)	53.47 (+40%)	82.73 (+8%)	49.51 (+38%)	0.0838(-16%)
	Added-8	**68.3 (+30%)**	**53.4 (+40%)**	**83.4 (+9%)**	**51.86 (+45%)**	0.0886(-12%)
	Added-10	66.17 (+26%)	50.43 (+32%)	81.93 (+7%)	49.45 (+38%)	0.0729(-27%)
T+GCN	Added-2	65.61 (+24%)	50.13 (+31%)	82.39 (+7%)	48.82 (+36 %)	0.1119(+12%)
	Added-4	67.92 (+29%)	52.97 (+39%)	83.23 (+8%)	51.43 (+44%)	0.1537(+53%)
	Added-6	**73.04 (+39%)**	**57.48 (+50%)**	**85.55 (+11%)**	**57.53 (+61%)**	0.1506(+50%)
	Added-8	71.49 (+36%)	54.33 (+42%)	84.9 (+11%)	55.63 (+55%)	0.155(+55%)
	Added-10	70.7 (+34%)	51.91 (+36%)	83.97 (+9%)	54.67 (+53%)	0.1584(+58%)
T+ BERT	Added-2	**55.99 (+6%)**	**40.77 (+7%)**	**77.26 (+1%)**	**38.88 (+9%)**	-0.0355(-67%)
	Added-4	48.47 (-8%)	32.18 (-16%)	73.82 (-4%)	31.99 (-11%)	-0.0134(-133%)
	Added-6	55.72 (+6%)	37.43 (-2%)	77.11 (0%)	38.62 (+8%)	-0.0103(-90%)
	Added-8	47.91 (-9%)	30.42 (-20%)	73.77 (-4%)	31.5 (-12 %)	-0.0392(-61%)
	Added-10	38.2 (-28%)	19.67 (-49%)	68.39 (-11%)	23.61 (-34%)	-0.0687(-32%)

each of the target hashtags. For T+HashtagPool, the performance in terms of F-Measure, Rand Index, and JC are 3%, 2%, and 5% lower than raw tweets, whereas NMI is increased by 6%. The decrease in performance of F-Measure, Rand Index and JC may be attributed to increase in the noise of the tweets after expanding it with the hashtags pool.

In case of Homogeneous dataset (as given in Table 3.6), LDA performance on raw tweets in terms of F-Measure, NMI, Rand Index, and JC are 52.72%, 38.20%, 76.75%, and 35.8% respectively. Using T+BiLSTM approach, all the setup except Added-8 hashtags perform better than raw tweet in terms of F-Measure, NMI, Rand Index, and JC. The best performance is observed at Added-2 hashtags with an improvement of 27%, 38%, 8% and 41% in terms of F-Measure, NMI, Rand Index, and JC. For T+1-hop N, all the setup except Added-2 hashtags perform better than the raw tweet. The performance continues to increase as we increase the number of hashtags added, and the maximum performance is observed at Added-8 hashtags with an improvement of 30%, 40%, 9%, and 45% in terms of F-Measure, NMI, Rand Index, and JC respectively. After attaining the maximum performance at Added-8 hashtags, it decreases for Added-10 hashtags. For T+GCN approach, all the setups performs better than the raw tweets. The performance continues to increase as we increase the number of added hashtag, and

the maximum performance is reached at Added-6 hashtag with an improvement of 39%, 50%, 11%, and 61% in terms of F-Measure, NMI, Rand Index, and JC respectively. After reaching at its maximum, the performance starts decreasing for Added-8 and Added-10 hashtags. For T+BERT approach, only Added-2 and Added-6 hashtag performs better than raw tweets. The best performance is observed at Added-2 hashtag with an improvement of 6%, 7%, 1% and 9% in terms of F-Measure, NMI, Rand Index and JC respectively. For T+HashtagPool approach, the performance as compared to raw tweet is improved by 13%, 12%, 2% and 18% in terms of F-Measure, NMI, Rand Index and JC. The performance is greater than all the setup of T+BERT approach but is lesser than the best performance of T+BiLSTM, T+1-hop N, and T+GCN approach.

We observe that LDA performance using proposed tweet expansion approaches improves compared to raw tweets in most setups from the above results. For network-based graphical tweet expansion such as 1-hop nearest neighbor and GCN-based tweet expansion approaches, expanding tweets with a moderate number of hashtags (Added-6, Added-8, Added-10) give a better LDA performance over both Heterogeneous and Homogeneous datasets. Similarly, for BiLSTM (text-based sequential model), tweets expanded with a moderate number of added hashtags (Added-8) over the Heterogeneous dataset and tweets expanded with a smaller number of hashtags (Added-2) over the Homogeneous dataset give a better LDA performance. For BERT (text-based sequential model), tweets expanded with a smaller number of added hashtags (Added-2) giver a better LDA performance over both Heterogeneous and Homogeneous datasets.

3.5.3 Comparison of different hashtag-based tweet expansion approaches

In this subsection, we present few examples of tweet expansion using different text-based sequential and network-based graphical approaches over the Homogeneous and Heterogeneous datasets. Table 3.7 and 3.8 show few examples of added hashtags using BiLSTM, 1-hop N, GCN and BERT based tweets expansion approach on Homogeneous and Heterogeneous dataset. First tweet in Table 3.7 is “GST (creates 50-50 situation) – its a loss or benefit for Real Estate - very confusing !! URL”. The BiLSTM based sequential approach captures the sequential information and hence is able to predict the related hashtags at both topic level (#gstwhat, #emerging) and document level (#realstateindia) for the first tweet. The 1-hop

TABLE 3.7: Examples of semantic expansion of Heterogeneous dataset using BiLSTM, 1-hop N, GCN and BERT-based approaches.
Note: Semantically related hashtags to the tweet and its associated class are in blue, and hashtags related to other classes are in red.

Topic	Raw Tweet	BiLSTM	1-hop N	GCN	BERT
GSTN	GST (creates 50-50 situation) – its a loss or benefit for Real Estate - very confusing !! URL	#gstwhat, #emerging, #mustread, #namo, #newsflash, #realestateindia, #unityofindia, #primeminister, #haryana, #nlhafta	#gst, #gstbill, #news, #transformingindia, #india, #, #biharelections, #cab, #til_now, #surgicalstrike	#risingstarindia, #auction, #cars, #getreal, #wedojallikatu, #mustread, #andhrapradesh, #punjabelections2017, #financebill,	#surgicalstrike, #surgicalstrikes, #gst, #biharelections, #nrc, #uriattack, #surgicalstrikesagainstpak, #uriattacks, #pathankot, #kashmirunrest, #parsi_community
GSTN	Assam Becomes First State To Pass Bill On GST URL	#breaking, #gst, #news, #gstbill, #modi, #toi, #narendramodi, #bjp, #live, #justin	#cab, #gst, #biharelections, #nrc, #gstbill, #biharelection2020, #assam, #news, #india, #voteagainstc	#agp, #grandalliancebihar, #parsi_community, #fear, #gstcouncil, #demonetisation, #nda4bihar, #mfnstatus, #surgicalstrike, #westbengal	#surgicalstrike, #surgicalstrikes, #gst, #biharelections, #nrc, #uriattack, #uriattacks, #surgicalstrikesagainstpak, #pathankot, #gstbill
CAB	Manipur will come under purview of Inner Line Permit (ILP)–to get exempted from Citizenship Amendment Bill (CAB), says Amit Shah in Lok Sabha #CAB	#citizenshipamendment-bill, #assam, #indiasupportscab, #samjhakya, #minorities, #gotit, #cabprotest, #citizenshipbill, #indiarejectscab, #muslims	#cab, #nrc, #biharelections, #gst, #biharelection2020, #cabbill, #cabprotest, #gstbill, #assam, #citi-zenshipamendmentbill2019	#370, #students, #constitutionbetrayed, #citizenshipamendment-bill2019, #kashmir, #rhetoricalquestion, #hindunation, #shashitharoor, #citizenshipofindia, #indiawelcomescab	#cab, #surgicalstrike, #gst, #kashmir, #surgicalstrikes, #biharelections, #biharelection2020, #pakistan, #gstbill, #india

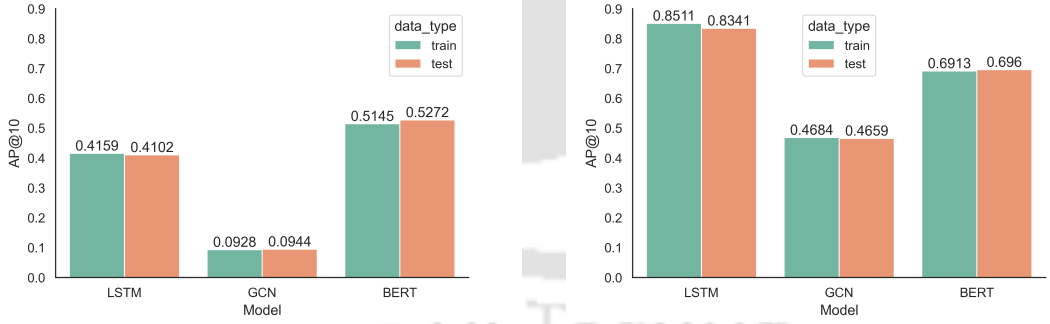
N based approach is based on word to word co-occurrences adjacency matrix, capturing the semantically related hashtags (#gst, #gstbill, #transformingindia) for the tweet, and also adding hashtags related to other classes (#biharlections, #cab, #surgical strike) causing a topic drift. The BERT-based approach adds only one related hashtag (#gst) for the tweet discussed above, and mostly hashtags from other classes.

In Table 3.8, the first tweet "RT @NewIndianXpress: #Pathankot hero's house to be partly razed; family's pleas fall on deaf ears URL", semantically related hashtags such as #pathankot, #pathankotattack, #bengaluru's, #bengaluru were added using BiLSTM, and #baramulla, #pathankot, #bangaluru, #niranjankumar is added using 1-hop N. Similarly, related hashtags #terrorstatepak, #pak were added using GCN and #pathankot, #indianarmy were added using BERT approach. The tweet is about demolition of the house of a Pathankot attack martyr. The BiLSTM based approach find location of martyr's house (#bengaluru, #bengaluru's) and the location related to the Pathankot attack (#punjab, #patahnkotattack, #pathankot). 1-hop N based approach also add the name of the martyr (#niranjankumar) along with the related hashtags found in BiLSTM based approach.

To quantify the performance of semantically related hashtags prediction by three models namely BiLSTM, BERT and GCN, we use Average Precision (AP@10)

TABLE 3.8: Examples of semantic expansion of Homogeneous dataset using BiLSTM, 1-hop N, GCN and BERT approach.
Note: Semantically related hashtags to the tweet and its associated class are in blue, and hashtags related to other classes are in red.

Topic	Raw Tweet	BiLSTM	1-hop N	GCN	BERT
Pathankot Attack	RT @NewIndianXpress: #Pathankot hero's house to be partly razed; family's pleas fall on deaf ears URL URL	#punjab, #pathankot, #pathankotattack, #bengaluru's, #stopfundingpakistan, #56inch, #this, #indiastrikesback, #bengaluru, #reform	#uri, #uriattack, #baramulla, #pathankot, #bengaluru, #surgicalstrike, #niranjankumar, #pakistan, #punjab, #india	#baramulla, #uriattack, #syriastrikes, #kashmir, #terrorstatepak, #unga, #india, #nawazsharif, #pak, #uri	#kashmir, #kashmirunrest, #uriattack, #pathankot, #kashmircrisis, #kashmirkillings, #indianarmy, #surgicalstrikesagainstpak, #burhanwani, #modipunishespak
Pathankot Attack	Home Ministry sanctions National Investigation Agency to prosecute terror group Jaish chief #MasoodAzhar, 3 others in #PathankotAttack: PTI	#masoodazhar, #ndtvbanned, #boycottpakornot, #news, #ndtv, #breakingnews, #india, #modi, #burhan, #syria	#surgicalstrike, #surgicalstrikes, #pathankot, #pakistan, #uri, #uriattack, #breaking, #pathankotattack, #masoodazhar, #indianarmy	#uriattack, #surgicalstrike, #pathankot, #loc, #uriattacks, #surgicalstrikesagainstpak, #modi, #gurdaspur, #letsdestroypak, #pakistan's	#kashmir, #uriattack, #kashmirunrest, #pathankot, #kashmircrisis, #baramulla, #pakistan, #kashmirkillings, #loc, #modipunishespak
Uri Attack	ndtv: Uri Brigade Commander shifted out. Court of inquiry underway in #UriAttack	#baramulla, #uri, #pak, #narendramodi, #indiastrikespak, #india, #orop, #india's, #jaihind, #pakistan's	#surgicalstrike, #surgicalstrikes, #baramulla, #modipunishespak, #pathankot, #uriattack, #pakistan, #pakartistsbanned, #kashmir, #indianarmy	#baramulla, #kashmir, #uriattack, #indianarmy, #india, #loc, #uri, #modi, #pakistan, #blackmoney	#kashmir, #uriattack, #kashmirunrest, #pathankot, #kashmircrisis, #pakistan, #kashmirkillings, #baramulla, #loc, #uri
Syria Crisis	Benjamin Netanyahu voices "total support" for Syria strikes #Damascus	#damascus, #israel, #maga, #russia, #yemen, #donaldtrump, #bringtroopshome, #syriastrike, #assad, #france	#surgicalstrike, #syriastrikes, #syria, #surgicalstrikes, #pakistan, #uri, #breaking, #uriattack, #syriastrike, #damascus	#syriastrikes, #syria, #kashmir, #unga, #india, #terrorstatepak, #news, #kashmir's, #syriastrike, #uri	#uriattack, #kashmirunrest, #indianarmy, #pathankot, #kashmircrisis, #kashmir, #surgicalstrike, #surgicalstrikesagainstpak, #kashmirkillings, #burhanwani



(A) AP@10 for Heterogeneous Dataset. (B) AP@10 for Homogeneous Dataset.

FIGURE 3.5: Evaluation of hashtag-based tweets expansion using BiLSTM, GCN and BERT model over Heterogeneous and Homogeneous Dataset in terms of Average Precision (AP@10).

metrics from the top-10 predicted hashtags. We define Average Precision out of top-10 predicted hashtags (AP@10) as follows:

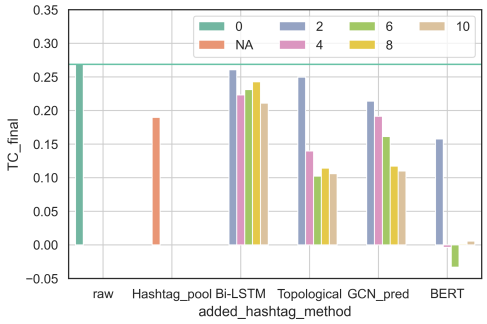
$$AP@10 = \frac{1}{N} \sum_{i \in [1,N]} (TP_i) \tag{3.5}$$

where TP_i is equal to one if the target hashtag is present in top-10 predicted hashtags and 0 otherwise, and N is the total number of the tweets. For the evaluation of related hashtags prediction, we have used the same training and testing as prepared in the subsection 3.3.1 and Table 3.2. We took out one hashtag one by one and used it as target hashtags. We use trained model of BiLSTM, BERT and GCN to predict the target hashtags. Figure 3.5 presents a comparative performance of the three models over Heterogeneous and Homogeneous dataset. From the figure, we observe that BERT and GCN have superior performance as compared to GCN model for both the datasets. Further, the performance of all three models (BiLSTM, BERT and GCN) over Homogeneous dataset is superior in case of Homogeneous dataset as compared to Heterogeneous dataset.

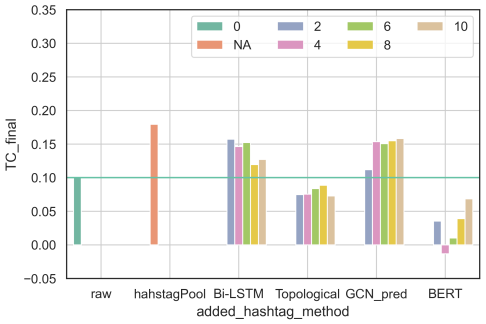
3.5.4 Topic quality comparison

This subsection presents the quantitative analysis (using topics coherence [55, 56]) and qualitative analysis (using top words, document support, tentative class label) of topics using different hashtag-based tweet expansion approaches. Figure 3.6 presents topic coherence and F-measure of raw tweets and expanded tweets using different approaches over Heterogeneous and Homogeneous dataset. From the figure, we observe that most of the setups of hashtag-based tweet expansion approaches (except BERT) performs better than raw tweet in terms of F-measure over Heterogeneous and Homogeneous dataset. Similarly, most of the hashtag-based tweet expansion approaches (except BERT and 1-hop N) performs better than raw tweets in terms of Topic coherence, whereas all hashtag-based tweet expansion approaches performs lesser than raw tweets in terms of F-measure. the lesser value of topic coherence in case of Heterogeneous dataset can be attributed to lesser dataset overlapping and more data-sparsity. We may need to formulate a better measure than word co-occurrence to estimate the relatedness of word in tweets.

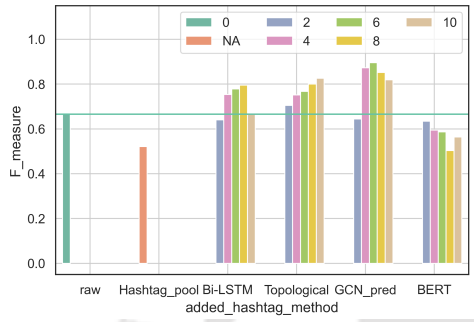
Further, Table 3.9 and 3.10 presents qualitative analysis of topics obtained by the best performing setups of different tweet expansion approaches T+BiLSTM, T+1-hop N, T+GCN, T+BERT, T+HashtagPool along with raw tweet in terms of F-Measure and NMI. In the literature [2, 72, 41], qualitative analysis of topic obtained by LDA is done using top words and tentative manually assigned class label. However, in our case, assigning a manual class label using top-10 words is a challenging task owing to overlapping of words and hashtags between classes. And, in absence of class label of the topics, the assessment of the topic quality



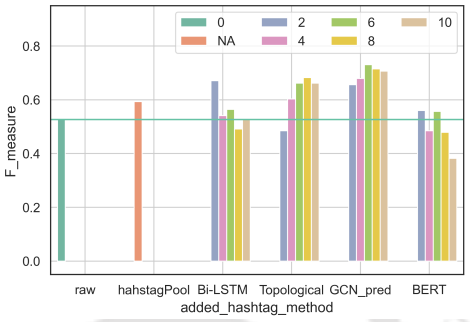
(A) TC for Heterogeneous Dataset.



(B) TC for Homogeneous Dataset.



(C) F-measure for Heterogeneous Dataset.



(D) F-measure for Homogeneous Dataset.

FIGURE 3.6: Topic Coherence (TC) and F-measure of LDA over raw tweets and different hashtags-based tweets expansion approaches using Heterogeneous and Homogeneous Dataset.

and human interpretability of topics between different approaches of hashtag-based tweet expansion becomes a challenging task. very challenging task. Therefore, to simplify the comparative assessment of the topic quality between different hashtag-based tweet expansion approaches, the following assumptions are made:

- Each document is assigned with the topic having the highest distribution in its doc-topic distribution θ_d.
- And for each topic, we give a tentative class label same as the class label of maximum documents belonging to the topic.

The above assumptions are only for simplification of topic quality analysis of different hashtag-based tweet expansion approaches. Using the above assumption, we assign a tentative class label to topics using document-topic distribution obtained by LDA and ground truth class label of documents. Further, we also measure the homogeneity of the topic using Document Support *(DS)* as follows:

$$DS(t) = \frac{N_t^c}{N_t} \tag{3.6}$$

where N_t denotes the number of documents assigned to topic t, and N_t^c denotes the number of document assigned to topic t and have class label c, and c is the tentative class label given to the topic t.

TABLE 3.9: Qualitative assessment of topics obtained by LDA over Heterogeneous dataset for raw tweet and different hashtag-based tweet expansions. **Note**: Hashtags related to manually assigned classes for each topic are in blue, and hashtags related to other classes are in red. DS stands for Document Support.

Model Name	Topic	Tentative class label	DS (%)	Top-words
Raw tweet (T)	topic-0	CAB protest	91.74	want, #cab, assam, nothing, passing, worry, brothers, assure, sisters, citizenship
	topic-1	GSTN	78.28	gst, will, bill, #gst, #cab, india, pm, tax, modi, sabha
	topic-2	BiharElection2020	95.57	#biharelections, bihar, #biharelection2020, bjp, will, phase, live, #biharpolls, seats, #bihar
	topic-3	Attack	71.08	#surgicalstrike, pathankot, burhan, u, indian, #cab, army, wani, india, #surgicalstrikes
T+ BiLSTM	topic-0	GSTN	90.81	gst, #gst, #gstbill, #transformingindia, #india, #modi, will, bill, #loksabha, #narendramodi
	topic-1	Attack	75.92	#surgicalstrike, #pakistan, #kashmir, #india, #indianarmy, pathankot, #uriattack, burhan, #baramulla, #loc
	topic-2	CAB protest	99.34	#cab, want, #citizenshipamendmentbill2019, #bjp, #cab2019, #indiasupportscab, assam, #cabbill2019, nothing, #citizenshipammendmentbill2019
	topic-3	BiharElection2020	95.08	#biharelections, #biharelection2020, #biharelections2020, #bihar, #biharpolls, bihar, #bjp, #nitishkumar, #voteonbihar, #biharwithnda
T+ 1-hop N	topic-0	Attack	81.84	#surgicalstrike, #cab, #surgicalstrikes, #biharelections, #uriattack, #biharelection2020, #gst, #baramulla, #pakistan, #indianarmy
	topic-1	GSTN	95.09	#gst, #gstbill, #cab, #biharelections, #india, #biharelection2020, gst, #surgicalstrike, #, #news
	topic-2	CAB protest	97.08	#cab, #nrc, #cabprotest, #cabbill, #citizenshipamendmentbill2019, #assam, #cab2019, #citizenshipamendmentbill, want, #biharelections
	topic-3	BiharElection2020	91.44	#biharelections, #biharelection2020, #biharpolls, #bihar, #biharelections2020, #voteonbihar, #biharwithnda, #nda, #cab, #bjp
T+ GCN	topic-0	CAB protest	96.74	#cab, want, #agp, #citizenshipamendmentbill, #citizenshipamendmentbill2019, #constitutionbetrayed, assam, nothing, passing, worry
	topic-1	GSTN	96.22	gst, #parsi_community, #asiacup, will, #gst, bill, #agp, #india, #nepal, #gstcouncil
	topic-2	Attack	90.80	#surgicalstrike, #uriattacks, #dontforgetpast, pathankot, burhan, #operationbadla, #electi, #pathankot, #india, #wetrustonmodi
	topic-3	BiharElection2020	95.47	#biharelections, #grandalliancebihar, #biharpolls, #biharrejectsnda, #biharelectionresults, #jdu, bihar, #voteonbihar, #hathras, bjp
T+ BERT	topic-0	BiharElection2020	89.30	#surgicalstrike, #surgicalstrikes, #biharelections, bihar, #biharelection2020, bjp, phase, live, amp
	topic-1	GSTN	81.08	#surgicalstrike, #surgicalstrikes, gst, #gst, bill, pm, tax, modi, india, amp
	topic-2	Attack	64.61	#surgicalstrike, #surgicalstrikes, amp, pathankot, burhan, #cab, u, india, indian, wani
	topic-3	CAB protest	94.10	want, #surgicalstrike, #surgicalstrikes, #cab, assam, nothing, passing, worry, brothers, assure
T+Hash tagPool	topic-0	GSTN	54.67	hon, #biharelections, pm, bjp, new, election, ji, meeting, union, committee
	topic-1	CAB	98.26	#cab, religious, bangladesh, pakistan, applies, afghanistan, #rohingya, m, fleeing, #myanmar
	topic-2	Attack	68.21	#surgicalstrike, #surgicalstrikes, #uriattack, indian, #kashmir, army, #kashmirunrest, india, #india, #pathankot
	topic-3	BiharElection2020	65.41	#biharelections, #biharelection2020, #bihar, #biharpolls, bihar, #biharelections2020, seats, ±, bjp, will

As presented in Table 3.9, topics given by LDA on raw tweets and different tweet expansion approaches using Heterogeneous dataset are well separable and represents all four classes present in the dataset namely: CAB protest, GSTN, BiharElection2020, and Attack. For CAB protest class, topic-0 of raw tweet is assigned with 91.74% of documents support (using the ground truth class label), which increases to 99.34% in case of topic-3 of T+BiLSTM approach, 97.08% in case of topic-2 of T+1-hop N, 96.74% topic-0 of T+GCN, 94.10% in case of T+GCN, and 98.26% in case of topic-1 of T+HashtagPool. For GSTN class, topic-1 of raw tweet

is assigned with 78.28% document support, which increases to 90.81% in case of topic-0 of T+BiLSTM, 95.09% in case of topic-1 of T+1-hop N, 96.22% in case of topic-1 of T+GCN, 81.08% in case of T+BERT, and decreases to 54.67% in case of topic-0 of T+HashtagPool. For BiharElection2020 class, topic-2 of raw tweet is assigned with 95.57% document support, which decreases to 95.08% in case of topic-3 of T+BiLSTM, 91.44% in case of topic-3 of T+1-hop N, 95.47% in case of topic-3 of T+GCN, 89.30% in case of topic-0 of T+BERT, and 65.41% in case of topic-3 of BiharElection2020. For Attack class, topic-3 of raw tweet is assigned with 71.08% of document support, which increase to 75.92% in case of topic-2 of T+BiLSTM, 81.84% in case of topic-0 of T+1-hop N, 90.80% in case of topic-2 of T+GCN, and decreases to 64.41% in case of topic-2 of T+BERT and 68.21% in case of topic-2 of T+HashtagPool.

Similar to Heterogeneous dataset, Table 3.10 presents top 10 words for each topic and tentatively topic label using LDA on raw tweets, and different tweet expansion approaches over Homogeneous dataset. Homogeneous dataset, as given in Table 3.3, contains five classes namely Surgical Strike (7585 tweets), Kashmir Unrest (5947 tweets), Pathankot Attack (5057 tweets), Syria Crisis (1012 tweets) and Uri attack (747 tweets) with high overlapping of hashtags and keywords between different classes. LDA on raw tweet fails to represent Syria Crisis and Uri Attack, classes with lesser number of tweets. Further, the dominant classes Pathankot Attack and Surgical Strike is represented by more than one topic. Majority of the documents belonging to Uri attack is merged with topic-0 (Pathankot Attack) and topic-1 (Surgical Strike). Similarly, the majority of documents belonging to Syria crisis is merged with topic-3 (Kashmir Unrest). Similar observation is reported in the study [72] for using LDA over Reuters corpus with skewed class distributions, where a dominant class is represented by multiple topics.

T+BiLSTM approach over Homogeneous dataset gives the representative topic for all classes except Uri Attack and gives two representative topics for Pathankot Attack. Majority of the documents belonging to Uri Attack is merged with topic-1 (Pathankot Attack). T+1-hop N approach gives the representative topic for all classes except Surgical Strike and Syria crisis and gives two representative for Surgical Strike. Majority of the documents belonging to Uri Attack is merged with topic-0 (Pathankot Attack). Similarly, most of the documents belonging to the Syria Crisis are merged with topic-4 (Kashmir Unrest). T+GCN approach gives the representative topic for the classes except Uri Attack and gives two representative topics for Pathankot Attack. Majority of the documents belonging to Uri Attack is merged with topic-0 (Pathankot Attack). Similarly, T+BERT

TABLE 3.10: Qualitative assessment of topics obtained by LDA over Homogeneous dataset for raw tweet and different hashtag-based tweet expansions. **Note**: Hashtags related to manually assigned classes for each topic are in blue, and hashtags related to other classes are in red. DS stands for Document Support.

Model Name	Topic	Tentative class label	DS (%)	Top-words
Raw tweet(T)	topic-0	Pathankot Attack	78.67	pathankot, uri, attack, pak, u, attacks, pakistan, india, 26/11, modi
	topic-1	Surgical Strike	65.74	#surgicalstrike, #surgicalstrikes, army, #uriattack, india, indian, #baramulla, pathankot, #indianarmy, #kashmirunrest
	topic-2	Surgical Strike	92.82	#surgicalstrike, indian, army, #surgicalstrikes, loc, pak, across, pakistan, india, pm
	topic-3	Kashmir Unrest	30.86	syria, attack, #kashmir, #surgicalstrike, pathankot, #kashmirunrest, indian, #kashmircrisis, s, #kashmirkillings
	topic-4	Kashmir Unrest	84.49	burhan, wani, u, terrorist, kashmir, like, son, ur, will, pak
T+ BiLSTM	topic-0	Surgical Strike	92.79	#surgicalstrike, #surgicalstrikes, indian, #pakistan, army, loc, #indianarmy, #modipunishespak, #indiastrikesback, #india
	topic-1	Pathankot Attack	65.56	pathankot, uri, attack, #baramulla, #uriattack, #surgicalstrike, pak, #india, u, #uri
	topic-2	Syria Crisis	55.12	syria, attack, army, indian, #surgicalstrike, #syriastrikes, #syria, #uriattack, pak, #surgicalstrike-killed
	topic-3	Pathankot Attack	71.73	pathankot, hai, ki, ko, ka, #pathankot, house, martyr, attack, demolition
	topic-4	Kashmir Unrest	93.40	burhan, wani, #kashmir, u, terrorist, #kashmirunrest, kashmir, #baramulla, #unga, #india
T+1-hop N	topic-0	Pathankot Attack	56.37	pathankot, #india, #baramulla, #pak, #uriattack, #uri, #pathankot, uri, #pakistan, #kashmir
	topic-1	Surgical Strike	88.51	#surgicalstrike, #pakistan, #modipunishespak, #uriattack, #modi, #india, #indianarmy, #indiastrikesback, #baramulla, #loc
	topic-2	Surgical Strike	77.82	#surgicalstrike, #surgicalstrikes, #uriattack, #indiastrikesback, #modipunishespak, #modi, #pakistan, army, #indianarmy, indian
	topic-3	Pathankot Attack	55.25	pathankot, burhan, uri, u, wani, attack, #backarmyendpolitics, modi, #presstitutes, n
	topic-4	Kashmir Unrest	75.40	burhan, #kashmir, wani, #unga, #syriastrikes, #baramulla, #india, #terrorstatepak, #freekashmir, #pak
T+ GCN	topic-0	Pathankot Attack	70.13	#uriattack, #uriattacks, pathankot, #pathankot, #pathankotattack, #burhanwani, #surgicalstrike, uri, #india, #surgicalstrikepolitics
	topic-1	Syria Crisis	80.15	#syria, #syriastrikes, syria, attack, #russia, #trump, #syriastrike, #uri, #damascus, #india
	topic-2	Surgical Strike	92.60	#surgicalstrike, #uriattack, #surgicalstrikes, #pakistan, #indianarmy, #loc, #modipunishespak, #indiastrikesback, indian, #uri
	topic-3	Pathankot Attack	75.29	pathankot, #uriattacks, #uriattack, #pathankot, #pathankotattack, #burhanwani, #india, #bengaluru, #surgicalstrikepolitics, #demolishedbycorruption
	topic-4	Kashmir Unrest	94.08	burhan, #kashmir, #burhanwani, #india, wani, #kashmirunrest, #kashmirkillings, #baramulla, #kashmircrisis, #pakistan
T+ BERT	topic-0	Pathankot Attack	61.49	#surgicalstrike, pathankot, #uriattack, #kashmirunrest, #surgicalstrikes, #kashmir, uri, amp, #indianarmy, attack
	topic-1	Surgical Strike	49.42	#surgicalstrike, #surgicalstrikes, #uriattack, #kashmirunrest, attack, syria, #kashmir, #indianarmy, indian, army
	topic-2	Kashmir Unrest	88.59	burhan, #kashmirunrest, #kashmir, wani, #surgicalstrike, #uriattack, #surgicalstrikes, u, #indianarmy, terrorist
	topic-3	Surgical Strike	90.93	#surgicalstrike, #surgicalstrikes, indian, army, #kashmirunrest, loc, #indianarmy, #uriattack, #kashmir, across
	topic-4	Pathankot Attack	68.39	pathankot, #kashmir, #uriattack, #kashmirunrest, #surgicalstrike, hai, #baramulla, ka, ko, ki
T+Hash tagPool	topic-0	Syria Crisis	57.88	#syria, #syriastrikes, syria, attack, #russia, war, #syriastrike, s, trump, russia
	topic-1	Pathankot Attack	69.27	#uriattack, #pathankot, #uri, #baramulla, pak, india, attack, pakistan, now, #surgicalstrike
	topic-2	Kashmir Unrest	65.52	#surgicalstrike, #indianarmy, #pakistan, #burhanwani, #india, #surgicalstrikes, #loc, #uriattacks, #kashmir, #uriattack
	topic-3	Surgical Strike	90.42	#surgicalstrike, #surgicalstrikes, indian, army, #uriattack, loc, india, pathankot, pm, pak
	topic-4	Kashmir Unrest	75.86	#kashmir, #kashmirunrest, #kashmirkillings, #kashmircrisis, indian, army, kashmir, #surgicalstrikesagainstpak, n, pak

gives representative topics for all classes except Surgical Strike and Uri Attack and gives two representative topics for Surgical Strike and Pathankot Attack. The majority of documents belonging to Uri Attack are merged with Topic-0 (Surgical Strike), and the majority of documents belonging to the Syria Crisis are merged

with topic-0 (Pathankot Attack). T+HashtagPool gives the representative topics for all classes except Uri Attack and gives two representative topics for Kashmir Unrest. Majority of the documents belonging to Uri attack is merged with topic-1 (Pathankot Attack).

3.6 Summary and Future work

This chapter proposes the expansion of tweets with semantically related hashtags using text-based and graph-based approaches to handle the sparsity and under-specificity of tweets. First, we evaluated the importance of hashtags in LDA performance over tweets. Experimental results of LDA over two datasets of distinct nature: Homogeneous dataset (classes with overlapping keywords and hashtags) and Heterogeneous datasets (classes with less overlapping of keywords and hashtags) using different setups: LDA over raw tweets, tweets without hashtags, tweets with mentions, tweets without mentions and keywords shows that hashtags are an important feature for finding topics. Furthermore, to expand tweet with semantically related hashtags, we explored BiLSTM and BERT based sequential model in the text-based approach to get the tweet representation using textual content. And, in the case of graph-based approach tweet expansion with semantically related hashtags, we explored 1-hop nearest neighbor and Graph Convolution Network *(GCN)* to model tweet representation using word co-occurrence graph. We have evaluated the efficacy of proposed tweet expansion by comparing the performance of LDA over expanded tweets compared to raw tweets. LDA performance after expanding tweets with the proposed expansion approaches improves significantly compared to raw tweet and hashtag pooling based tweets expansion. The results show that the percentage of improvement after tweet expansion is more in the Homogeneous dataset than the Heterogeneous dataset when compared to the raw tweets. Further, the proposed tweet expansion methods also perform better in finding distinct topic representation of classes with less document support.

The future exploration of the chapter can be broadly categorized into two parts: a) hashtag-based tweets expansion module, and b) Topic modeling module. In hashtag-based tweet expansion module, we would like to experiment with different word embedding approaches (GloVe, word2vec, FastText) and the effect of attention to find the semantically related hashtags for a tweet. We would also like to study the effect of the tweet creation time in hashtag-based tweet expansion module. In the topic modeling module, we would like to study the impact

of different topic models such as (Biterm Topic Model (BTM) [41], Embedding-based Topic Model (ETM) [133], Topic modeling in embedding spaces [56], and tBert [134]) over hashtag-based expanded tweets. Furthermore, we would also like to explore the effect of different deep learning-based classifier [135] and supervised topic models such as Labeled LDA [64] over hashtag-based expanded tweets.

Chapter 4

Prioritizing Hashtags for Improved Topic Modeling over Tweets

The previous chapter proposes hashtag based tweets expansion for improved topic modeling over tweets. From experimental results, it is observed that hashtags-based tweet expansion improves the topic modeling performance by addressing the under specificity and data sparsity. Traditional LDA uses symmetric Dirichlet prior over topic-word Multinomial, giving equal importance to all the words in a document. This chapter proposes to utilize some special words or tokens such as hashtags by giving them more weights (priorities) over other words.

Similar to chapter 3, this chapter also harnesses the hashtags' efficacy in topic modeling by prioritizing them over other words. Previous studies such as Seeded-LDA [72] have considered similar assumption to guide LDA in discovering topics as per user belief. In this chapter, we propose Hashtag Prioritized LDA *(HP-LDA)* to guide LDA in connecting tweets to the underlying topics. Further, we extend HP-LDA as Prioritized Named Entity driven LDA *(PNE-LDA)* to study and analyze the effect of prioritizing the named entities in three news datasets.

4.1 Introduction

Traditional LDA [2] finds topics in large text collections based on word co-occurrences at document level by giving equal importance to all the words and makes no assumption about the underlying topics. However, in many real-world scenarios, some information about the underlying dataset/topics is known to users. For example, if we apply topic modeling over tweet collection, tokens like hashtags or mentions may have different importance than other texts. LDA fails to find proper topics in short and noisy text collections [74, 35, 31], skewed topic distributions [39], and texts with overlapping vocabularies. As reported in the studies [31, 32], topic modeling on tweets poses many challenges due to various issues like data sparsity, under-specificity, multilingual content, textual noises, etc. Pooling of tweets by common hashtags [35] or users [31], augmenting tweets with related contents from external sources [85, 33, 34] are some earlier approaches to address under-specificity in tweets. While the above studies modify the tweet content before applying topic modeling, authors in [40, 62] exploit meta information such as user's profile, user's activities, location etc. in discovering topics. Similarly, authors in [32] exploit the user-topic relationship to enhance topic modeling.

Studies [43, 44, 45] show that hashtags often provide useful information linking a tweet to its underlying topics, as they are provided by the person who posted the tweets. Similarly, hashtags of tweets posted during a time span may be related to popular events happened during that time. It motivates two approaches; (i) incorporate token importance in topic modeling, and (ii) assume to know few representative words of the underlying topics. Seeded-LDA [72] is one such method that assumes to know few words representing the topics. In this chapter, we consider token importance and propose a variant of LDA, which assigns different weights to different tokens and guide the topic modeling process in LDA. Unlike Seeded-LDA [72], this chapter makes no prior assumption about the underlying topics, and the token importance is estimated globally from the corpus.

Motivated by the observations [43, 44, 45] regarding the role of hashtags in connection to the underlying topic, this chapter proposes Hashtag Prioritized LDA *(HP-LDA)* which considers hashtags as more important tokens and prioritizes them over other tokens (keywords or mentions). The prioritized hashtags guide topic modeling process in LDA. Though the proposed HP-LDA can prioritize a set of any token in general, this work investigates the effect of hashtags prioritization. From various experimental setups over two types of datasets of different natures; (i) Heterogeneous- tweets collected from dissimilar topics and (ii) Homogeneous –

tweets collected from similar topics.It is evident that the proposed HP-LDA not only discover topics better than traditional LDA, but also can handle better in highly overlapping scenarios.

Similar to the role of hashtags in case of tweets, named entities are major topic/event descriptive terms in case of news articles [136]. In this chapter, we also extended the HP-LDA as Prioritized Named Entity driven LDA *(PNE-LDA)* to study and analyze the effect of prioritizing the named entities in three news datasets namely Bomb Blast, tweetReuters-21578-R [1], and 20-Newsgroup [2]. We observed that PNE-LDA outperforms LDA and Seeded-LDA for entity-driven topics.

4.1.1 Contribution

The key contributions of this chapter are:

- Studied effect of different tokens (hashtags, keywords, mentions) on topic modeling using LDA,
- Proposed Hashtag Prioritized LDA *(HP-LDA)* to incorporate different weights assigned to different tokens.
- Compared different prioritization strategies to estimate weights of hashtags in HP-LDA.
- Extended the HP-LDA as Prioritized Named Entity driven LDA *(PNE-LDA)* to study and analyze the effect of prioritizing the named entities in three news datasets namely Bomb Blast, Reuters-21578-R, and 20-Newsgroup and compared the performance with LDA [2] and Seeded-LDA [72].

4.2 Related work

In the past, several studies improve the topic modeling performance by utilizing the different meta information such as location information, document publishing time information in case of news media [62], user's profile information, user's tweets and re-tweet count, tweet location and bursty keyword information in case

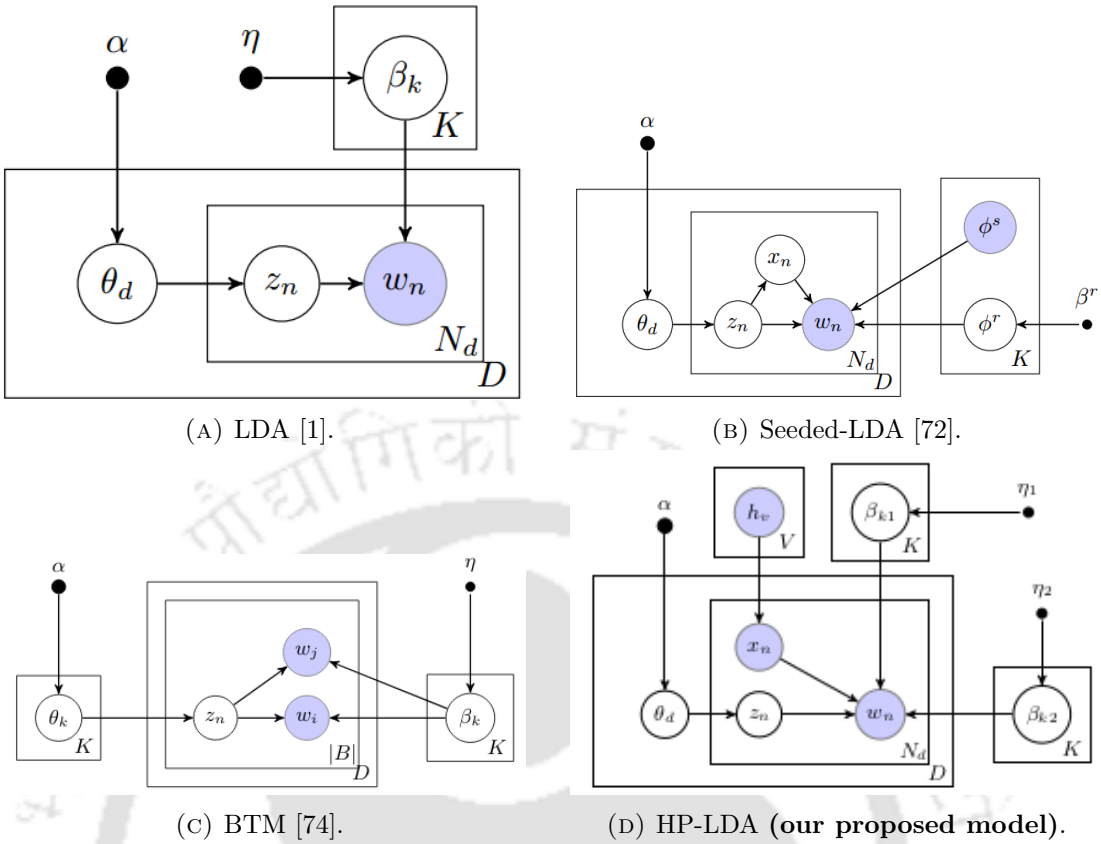
[1]https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection
[2]https://archive.ics.uci.edu/ml/datasets/Twenty+Newsgroups

of tweets [73, 40]. Pan et al. [62] proposed SpaceTimeLDA to incorporate location information and document publishing time information into LDA to detect event from TDT3 [63] and Reuters news corpus with an intuition that different reporting of same event share location and temporal information. Diao et al. proposed TimeUserLDA [73] to detect bursty topics on the Twitter dataset by incorporating tweet posting time and user timeline activity information into LDA. Authors in [73] assume that each tweet contains only one topic, and topic distribution of a tweet is either dependent on user personal interest(local topics) or on timestamp (global topics). Similarly, a word is sampled for every tweet, either from a Multinomial of background words or topic-word distribution. The authors reported improvement in burst topic detection performance in terms of Preciosion@5 compared to LDA, and other two variants of TimeUserLDA. Tsolmon et al. [40] proposed TimeReliableUser LDA to detect event incorporating the word weights based on time and user weights based on activity (weekly tweet and re-tweet count) and user popularity in tweet network. The authors reported improvement in event detection performance over LDA [2] and TimeUserLDA [73] over Korean tweets. Zhao et al. [32] proposed Twitter-LDA extending Author Topic model assuming tweet to have a topic distribution over user and all the words of a tweet to have a single topic. However, all the additional information used in the above studies are not always available, especially with the publicly available datasets. For example, only a small percentage of tweets are geotagged [42] and predicting the location of non geotagged tweets is a challenging problem. Similarly, collecting all the tweets of a user to estimate user-topic distribution is limited by Twitter API rate limit[3].

Yan et al. in [74] proposed Biterm Topic Model *(BTM)* to handle document wise word sparsity in short text, by modeling a global corpus-specific topic distribution (*theta*) instead of modeling document-specific topic distribution θ_d. Further, BTM utilizes word co-occurrence pattern by sampling bi-terms instead of sampling an unigram for every document as in LDA. The paper shows improved performance of BTM over LDA in terms of topic coherence and H-score (ratio of intra-cluster distance to inter-cluster distance) over Tweet-2011 used in TREC-2011 microblog task [4]. Wang et al. in [43, 75] propose an extension of LDA named as Hashtag Graph-based Topic Model *(HGTM)* to handle short text tweet sparsity by harnessing the hashtag-hashtag relation based on tweet co-occurrences. The HGTM model assigns a hashtag and topic pair for every word of a tweet. The authors reported improved performance of HGTM over LDA and other topic models such

[3] https://developer.twitter.com/en/docs/twitter-api/v1/tweets/timelines/overview

[4] https://trec.nist.gov/data/tweets/



(A) LDA [1]. (B) Seeded-LDA [72].

(C) BTM [74]. (D) HP-LDA **(our proposed model)**.

FIGURE 4.1: Plate diagram of LDA, Seeded-LDA, BTM and the proposed Hashtag Prioritized LDA *(HP-LDA)*

.

as Author Topic *(AT)* model, Latent Semantic Analysis in terms of H-score over Tweet-2011 datasets. Xing et al. [76] proposed hashtag-based Mutually Generative LDA *(MGe-LDA)* for sub-event discovery in tweet collection utilizing the hashtags. In MGe-LDA, both hashtag and topic mutually generate each other to mine the relationship between hashtags and topics. The authors reported an improved H-score over three sub-event from Tweet-2011 collection in H-score compared to LDA, HGTM and Author Topic Model. Similar to our approach, Jagarlamudi et al. [72] proposed Seeded-LDA.

4.3 Methodology

This section presents details of our proposed method Hashtag Prioritized LDA *(HP-LDA)* and compares the graphical plate diagram of related model such as LDA [2], BTM [74] and Seeded-LDA [72]. Further, we discuss the different hashtag prioritization approaches used in HP-LDA.

4.3.1 Hashtag Prioritized LDA (HP-LDA)

In case of regular text with skewed class distribution, LDA fails to represent original class distribution [72]. Further, LDA fails to learn suitable topics from short and noisy tweets [122, 74]. To tackle the above problem, we proposed Hashtag Prioritized LDA *(HP-LDA)*, which finds the topics using a set of prioritized hashtags over short and noisy tweets. In HP-LDA, we have considered a set of prioritized hashtags (h_v) to guide the topic inference process, similar to the original class distribution. In contrast to LDA, topic-word layer of HP-LDA is divided into two parts: a)topic-prioritized hashtag distributions (the model can be generalized using any words) and b)topic general words distributions, which enables the proposed algorithm to learn topics for prioritized words separately despite its low occurrences.

The graphical plate diagram of HP-LDA is given in Figure 4.1d. Each of the node in the plate diagram, in the Figure 4.1d, represents a random variable and an edge encodes probabilistic relationship between nodes. And each plate represent the multiple instance of same random variable class following similar relationship with other random variables. There are two types of random variables in the HP-LDA plate diagram : a)Observed random variables (shaded with light blue background color), and b) Non-observed random variables (with white background color). The description of different random variables used in plate diagram of HP-LDA are given in the table 4.1.

Table 4.1: HP-LDA parameter explanation

Parameter Name	**Symbol**	**Details**
Alpha	α	Doc-topic Dirichlet distribution
Eta 1	η_1	Topic-prioritized word Dirichlet distribution
Eta 2	η_2	Topic-general word Dirichlet distribution
theta d	θ_d	Doc-topic Multinomial parameter
Beta 1	β_1	Topic-prioritized word Multinomial parameter
Beta 2	β_2	Topic-general word Multinomial parameter
V	V	Total no of unique vocabulary
K	K	Total no of Topic
w_{dn}	w_{dn}	word of n^{th} word of d^{th} document
z_{dn}	z_{dn}	topic of n^{th} word of d^{th} document
w_{dn}	w_{dn}	word of n^{th} word of d^{th} document
x_{dn}	x_{dn}	switch variable corresponding to w_nd to indicate whether w_nd is prioritized word or not.

Generative algorithm of HP-LDA is present in algorithm 4. At first, the user provides a set of prioritized hashtags d_v. For every topic k, a topic-prioritized hashtag distribution and topic-general word distribution is generated using Dirichlet prior η_1 and, η_2 respectively. Further, for every document, a document-topic distribution (θ_d) is sampled using Dirichlet prior α similar to LDA. For every word of a document, a topic z_{dn} is sampled using Multinomial distribution θ_d and an observed switch variable x_{dn} is obtained using d_v, which tells whether a word is a prioritized hashtag or not. If x_{dn} is 0, we sample the word using topic-prioritized hashtag Multinomial distribution $\beta_{1z_{dn}}$, otherwise we sample the word using topic-general word Multinomial distribution $\beta_{2z_{dn}}$.

Comparison of graphical plate diagram of HP-LDA with related topic models such as LDA [2], BTM [74], and Seeded-LDA [72] is shown in Figure 4.1. In BTM model, the document-topic distribution is global and models a pair of bi-grams for every topic to model the word co-occurrences explicitly. BTM does not provide the individual topic distribution explicitly but can be found out by aggregating the topics assigned to bi-grams in the document. On the other hands, both HP-LDA and Seeded-LDA uses the set of prioritized tokens to model the topics as user's belief. In case of HP-LDA, user does not need to provide the topic label of prioritized word, and set of prioritized words and general words are disjoint. In contrast, user needs to provide topic label of prioritized word in case of Seeded-LDA and set of prioritized words and general words overlap.

The conditional probability of assigning a topic j to a word w_{dn} of document d by HP-LDA can be written as:

$$P(z_{dn}=j|z_{\neg nd},w_{\neg nd}) \propto \begin{cases} (\alpha + n^{w_d}_{\neg nd,j}) \dfrac{\eta_1 + n^{(w_{dn})}_{\neg nd,j}}{V_1.\eta_1 + n^{(.)}_{\neg nd,j}}, & \text{if } x_{dn}=0 \\ (\alpha + n^{w_d}_{\neg nd,j}) \dfrac{\eta_2 + n^{(w_{dn})}_{\neg nd,j}}{V_2.\eta_2 + n^{(.)}_{\neg nd,j}}, & \text{Otherwise} \end{cases} \tag{4.1}$$

where $x_{dn}=0$ indicates prioritized words and $x_{dn}=1$ indicates general words. The various symbols used in equation 4.1 are as follows:

- w_{dn} represents word at n^{th} index of document d
- z_{dn} represents topic of the word at n^{th} index of document d.
- $z_{\neg nd}$ represents all topics-word assignment except the current word topic assignment.

- $w_{\neg nd}$ represents all words in the vocabulary except the current word.
- $n^{w_d}_{\neg nd,j}$ represents number of words of current document assigned to the current topic j except the current word w_{dn}.
- $n^{(w_{dn})}_{\neg nd,j}$ represents number of words assigned to current topic j and similar to current word, except current word w_{dn}.
- $n^{(.)}_{\neg nd,j} = \sum_{\forall w_{dn} \in V} n^{(w_{dn})}_{\neg nd,j}$ represents number of words assigned to current topic j except current word w_{dn}.
- $V1$ and $V2$ represent the number of vocabulary of prioritized hashtags and general words.

In equation 4.1, the left-hand side of the equation $p(z_{dn} = j)$ resembles the probability of getting a topic j for word at n^{th} the index of the d_{th} document. The first term of the right-hand side equation resembles the probability of choosing a topic j from a Multinomial distribution of topics in the d_{th} document. The second term of the right-hand side of the equation refers to choosing a word w_{dn} from the topic j. If the word is a prioritized word, we sample it from prioritized hashtag-topic distribution parameterized by η_1; otherwise, we sample it from general word-topic distribution parameterized by η_2.

4.3.1.1 Different approaches used for hashtag prioritization used in HP-LDA

In this work, we have experimented with different approaches for selecting prioritized hashtags: a) manually selected hashtags, b) all hashtags c) prominent hashtags based on network centrality score over tweet word co-occurrences graph such as betweenness centrality, closeness centrality, degree centrality, and page rank centrality. A centrality measure captures the importance of a node in a network [137], some popular centrality measures used in social networks are degree centrality, closeness centrality, page rank centrality, and betweenness centrality. Degree centrality of a node is the ratio of direct link of the node to all the possible links in the network. Closeness centrality tries to capture how close a node is to any other node in the network, how quickly or easily can the node reach each other in the network. Betweenness centrality tries to capture the node role as a bridge or connected between other groups of nodes. PageRank Centrality is based on the PageRank value of the nodes in a graph – essentially, a node's importance based on its important neighbors which are highly linked.

ALGORITHM 4: The generative algorithm of HP-LDA. PHD represents Prioritized Hashtag Distribution and GWD represents General Words distribution.

// Generating distribution of word in Topics
for *each topic k in [1,K] :* **do**
    Generate Topic-PHD: β_{1k} using Dir(η_1)
    Generate Topic-GWD: β_{2k} using Dir(η_2)
for *each document d in Corpus:* **do**
    // Document generation
    Sample a topic distribution θ_d using $Dir(\alpha)$
    for *each of word w in document d :* **do**
        Sample a topic Z_{nd} using Multinomial(θ_d)
        Get x_{nd} = d[w_{nd}]
        if x_{nd} *equal to 0* **then**
            // Sample from Topic-PHD
            $w_{nd} \sim$ Multinomial(β_{1zdn})
        else if x_{nd} *equal to 1* **then**
            // Sample from Topic-GWD
            $w_{nd} \sim$ Multinomial(β_{2zdn})

TABLE 4.2: Heterogeneous dataset description.

S.No	Class Name	# of tweets	# of tweets with hashtag
1	GSTN	22512	7135
2	Attack	19336	12098
3	CAB protest	18434	18434
4	BiharElection2020	15600	15600

4.4 Results of HP-LDA

In this section, we discuss the tweet datasets characteristics used for HP-LDA, experimental setup, influence of hashtags in LDA over tweet, and comparative results of HP-LDA and its counterparts.

4.4.1 Datasets used for HP-LDA over tweets

For HP-LDA, we consider two types of tweet collections; (i) Heterogeneous – tweets collected from dissimilar topics, and (ii) Homogeneous – tweets collected from similar topics. Further, in Homogeneous datasets, we consider two distinct types of

Table 4.3: Attack dataset description.

S.No	Class Name	# of tweets	# of tweets with hashtag
1	Surgical Strike	7585	7543
2	Kashmir Unrest	5947	2361
3	Pathankot Attack	5057	1458
4	Syria Crisis	1012	408
5	Uri Attack	747	736

Table 4.4: Election dataset description.

S.No	Class Name	# of tweets	# of tweets with hashtag
1	WestBengalElection2021	17572	17572
2	BiharElection2020	15600	15600
3	AssamElection2021	9815	9815

topics namely Attack and Elections resulting in three experimental datasets. Tables 4.2, 4.3, and 4.4 show the characteristics of the Heterogeneous, Attack dataset, and Election dataset respectively. Heterogeneous dataset has 75,882 tweets with 4 distinct classes namely a) Goods and Services Tax Network *(GSTN)*, b) Attack c) Citizenship Amendment Bill *(CAB)*, and d) BiharElection2020. After removing user mentions, English stopwords, URLs, punctuation and emoticons, the Heterogeneous dataset is left with a vocabulary of 46,357 words, out of which there are 7,670 unique hashtags and 38,687 unique keywords. And, Attack dataset consist of total 20,348 tweets distributed under 5 similar topics namely: Uri Attack, Pathankot Attack, Kashmir Unrest, Surgical Strike, and Syria Crisis. After removing the user mention, English stop words, URLs, punctuation and emoticons, the Attack dataset is left with a vocabulary size of 21,701 words, out of which there are 2,617 unique hashtags and 19,084 unique keywords. And, Election dataset consist of total 42,987 tweets distributed under 3 similar topics namely: WestBengalElection2021, BiharElection2020, and AssamElection2021 respectively. The vocabulary size of the Election dataset is 20,571 words, out of which there are 3,429 unique hashtags and 17,142 unique keywords.

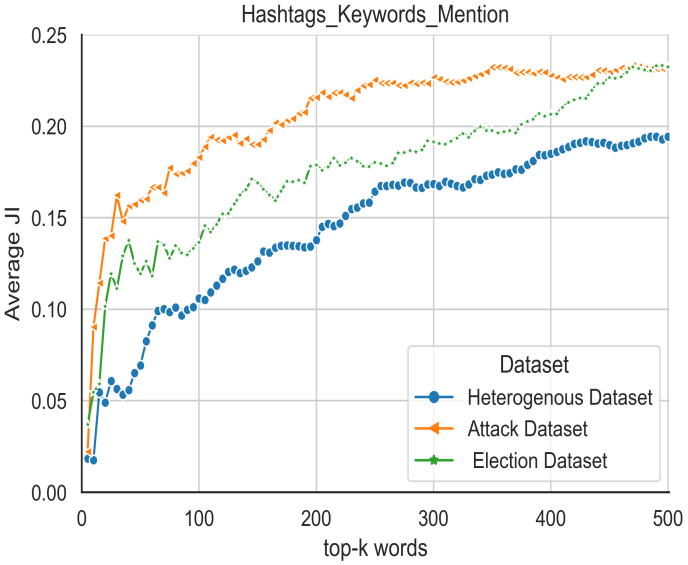


FIGURE 4.2: Measuring Heterogeneous, Attack and Election dataset overlapping in terms of Hashtags, Keywords, and Mentions using Average Jaccard Index (JI).

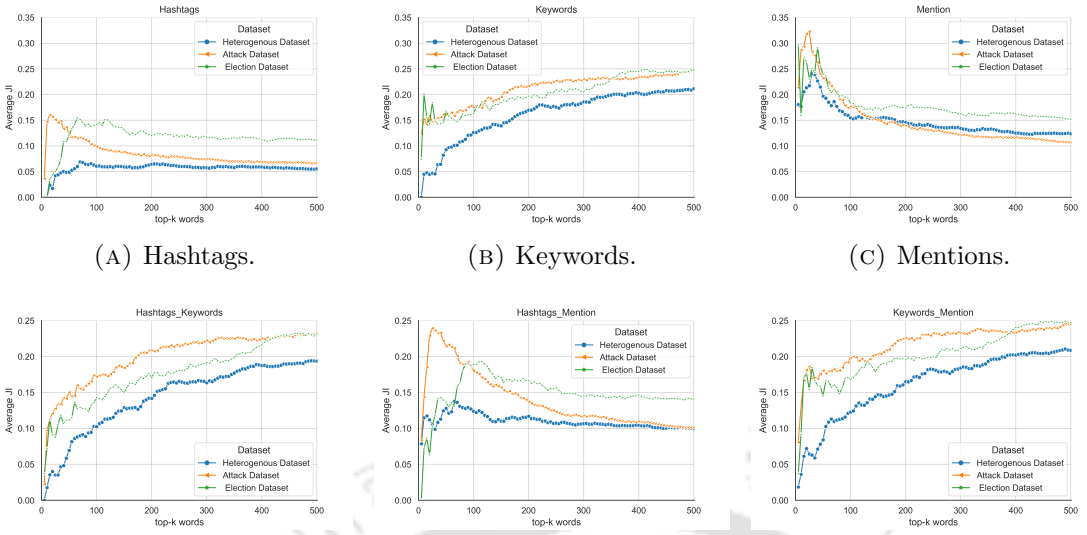
4.4.2 Analysis of hashtags, keywords and mentions overlapping in tweets datasets

We analyze the hashtags, keywords, and mentions overlapping across different topics using average Jaccard Index similarity between class pairs, similar to subsection 3.4.1 of the Chapter-3. Jaccard Index similarity between any two classes C_i and C_j considering `top-k` words for each of the classes can be defined as follows:

$$JI^k(C_i, C_j) = \frac{|S_i^k \cap S_j^k|}{|S_i^k \cup S_j^k|}. \tag{4.2}$$

where S_i^k and S_j^k represents the set of `top-k` words present in class label C_i and C_j respectively and k $\in \mathbb{N}$. Figure 4.2 present the datasets overlapping using average Jaccard Index similarity between class pairs using hashtags, keywords, and user mentions. From the figure, we observe that the Attack dataset has highest overlapping of words, followed by the Election dataset, whereas the Heterogeneous dataset has least overlapping of words. Moreover, the overlapping between class pair for all the three datasets increases as we increase the number of top-k words for each class.

Further, Figure 4.3 presents the overlapping of the three datasets using different combination of word types such as Keywords, Hashtags, Mentions, Hashtags and



(A) Hashtags. (B) Keywords. (C) Mentions.

(D) Hashtags and Keywords. (E) Hashtags and Mentions. (F) Keywords and Mentions.

FIGURE 4.3: Dataset overlapping using combination of word types of a tweet (Hashtags, Keywords, Mentions).

Keywords, Hashtags and Mentions, and Keywords and Mentions. From the figure, we observe that the Heterogeneous dataset have least overlapping compared to the Election and Attack datasets using different combination of word types. Moreover, when we consider only top-100 words for each class, the order of average JC over all the three datasets using only hashtags, only keywords, and only user mentions are as follows : i) average JC using only user mentions (¿0.18) ii) average JC using only Keywords (¡0.20) ii) average JC using only hashtags (¡0.10). From the above observation, we may conclude that user mentions are the most noisy features and hashtags are the best feature to distinguish between different classes. For the Hashtags, Mentions, Hashtags and Mentions, the average JC decrease for all the three datasets keeps on decreasing as we increase the value to top-k words for each class. For the Keywords, Hahstags and Keywords, Keywords and Mentions, the average JC decrease for all the three datasets keeps on increasing as we increase the value to top-k words for each class.

4.4.3 Experimental set up for HP-LDA and its counterparts

We perform prepossessing of the tweets using NLTK tweet tokenizer library[5]. We first remove the URLs, punctuations, and emoticons from the tweet text. We also

[5] https://www.nltk.org/

remove the user mentions and convert all the words into lowercase. We select a hashtag as prioritized word. Following are the different experimental setups:

- **LDA (Baseline)** [2]: We set document-topic Dirichlet parameter $\alpha = 0.1$ and η (topic-word distribution) as 0.3 for all three datasets.
- **BTM** [74]: We set the document-topic Dirichlet parameter $\alpha = 0.1$ and η (topic-word distribution) of Biterm Topic Model *(BTM)* as 0.3 same as LDA.
- **Hashtag Prioritized LDA (HP-LDA)**: We have considered hashtags as prioritized words. We set document-topic Dirichlet parameter $\alpha = 0.1$ for all three datasets. For Heterogeneous dataset, we set topic-prioritized word Dirichlet parameter $\beta_1 = 0.2$ and topic-non prioritized word Dirichlet parameter $\beta_2 = 0.3$. For Attack and Election dataset, we set topic-prioritized word Dirichlet parameter $\beta_1 = 0.1$ and topic non-prioritized word Dirichlet parameter $\beta_2 = 0.3$. We consider the following ways of selecting prioritized keywords : a) manually selected hashtags, b) all hashtags, c) betweenness centrality *(betC)*, d) closeness centrality Centrality *(closC)*, e) degree centrality *(degC)*, and f) page rank centrality *(prC)*. For all the centrality measure, we experimented with top 25%, 50%, and 75% of total hashtags based on the centrality score. To calculate the network centrality score of hashtags, we constructed a tweet graph using word co-occurrences matrix after removing stopwords.
- **Seeded-LDA** [72]: We set parameters of Seeded-LDA same as HP-LDA. We experiment with manually selected hashtags as prioritized word. Seeded LDA needs the mapping of the prioritized keywords with the class, whereas the hashtag selection methods (all hashtags, betC, degC, prC) only provides the important keywords based on the score but not the mapping with original class distribution). Hence, seeded LDA is experimented with manually selected hashtags.

We set the number of topics equal to the number of class in the Homogeneous, Election, and Attack datasets respectively. We run the collapsed Gibbs sampling for LDA and HP-LDA for learning the model up to 200 iterations.

TABLE 4.5: Effect of different entities combination using LDA performance in tweet over Heterogeneous, Attack, and Election dataset. WT-H stands for without hashtags, WT-M stands for without mentions, and WT-H-M stands for without hashtags, and mentions.

Dataset	Setup	F-Measure (%)	NMI(%)	Rand Index (%)	JC(%)
Heterogeneous dataset	raw tweet	49.36	34.64	73.65	32.77
	WT-H	**47.38% (-4%)**	**29.23 (-16%)**	**72.44 (-2%)**	**31.05 (-5%)**
	WT-M	60.25 (+22%)	47.93 (+38%)	78.63 (+7%)	43.11 (+32%)
	WT-H-M	52.24 (+6%)	35.93 (+4%)	74.44 (+1%)	35.36 (+8%)
Attack dataset	raw tweet	52.08	40.08	76.0	35.21
	WT-H	**49.64 (-5%)**	**32.61 (-19%)**	**74.14 (-3%)**	**33.01 (-6%)**
	WT-M	49.23 (-5%)	37.186 (-7%)	74.78 (-2%)	32.66 (-7%)
	WT-H-M	48.12 (-8%)	30.70 (-23%)	73.54 (-3%)	31.69 (-10%)
Election dataset	raw tweet	52.72	23.48	67.05	35.8
	WT-H	**38.99 (-26%)**	**5.84 (-75%)**	**57.21 (-15%)**	**24.22 (-32%)**
	WT-M	52.08 (-1%)	24.42 (+4%)	66.96 (0%)	35.2 (-2%)
	WT-H-M	38.67 (-27%)	6.75 (-71%)	57.86 (-14%)	23.97 (-33%)

4.4.4 Effect of different word type in LDA performance over tweets

Table 4.5 presents the effect of different word types (hashtags, keywords, mentions) in LDA performance over Heterogeneous, Election and Attack datasets. We study the influence of hashtags by performing the following analysis; i) perform LDA over raw tweets, ii) remove hashtags from the raw tweets and perform LDA (WT-H), iii) remove mentions from the raw tweet and perform LDA (WT-M), and iv) remove hashtags and mentions from the raw tweet (WT-H-M) and perform LDA. In case of Heterogeneous dataset, performance after removing hashtags is decreased by 4%, 16%, 2% and 5% in terms of F-Measure, Rand Index, Normalized Mutual Information *(NMI)* and Jaccard Coefficient *(JC)* respectively. The performance after removing mentions is increased by 22%, 38%, 7% and 32% in terms of F-Measure, NMI, Rand Index, and JC respectively. The performance after removing both hashtags and mentions is increased by 6%, 4%, 1% and 8% respectively. Therefore, hashtag is the most informative feature in case of heterogeneous dataset, as the LDA performance drop is maximum after removing hashtags.

In case of Attack dataset, the performance after removing hashtags is decreased by 5%, 19%, 3% and 6% in terms of F-Measure, NMI, Rand Index and JC respectively. The performance after removing mentions is decreased by 5%, 7%, 2%, and 7% in terms of F-Measure, NMI, Rand Index, and JC respectively. The performance after removing both hashtags and mentions is decreased by 8%, 23%, 3%, and

TABLE 4.6: Comparative results of LDA, BTM, Seeded-LDA and Hashtag Prioritized LDA(HP-LDA) over Heterogeneous dataset.

Method	Prioritization approach	% of hashtags used	F-Measure(%)	JC (%)	Rand Index(%)	NMI (%)
LDA [2]	-	-	68.61	52.22	83.38	58.46
BTM [74]	-	-	62.91 (-8%)	45.89 (-12%)	75.69 (-9%)	59.66 (+2%)
Seeded-LDA [72]	Manual hashtags	-	**76.75 (+12%)**	**62.28 (+19%)**	**88.01 (+6%)**	**68.25 (+17%)**
HP-LDA	Manual hashtags	-	**74.93 (+9%)**	**59.91 (+15%)**	**87.10 (+4%)**	**65.79 (+13%)**
	all_hashtags	100	**72.53 (+6%)**	**56.9 (+9%)**	**85.85 (+3%)**	**63.78 (+9%)**
	betC	25	71.97 (+5%)	56.22 (+8%)	85.56 (+3%)	62.52 (+7%)
		50	60.02 (-13%)	42.88 (-18%)	79.32 (-5%)	51.64 (-12%)
		75	**73.34 (7%)**	**57.91 (11%)**	**86.29 (3%)**	**63.19 (8%)**
	closC	25	69.73 (+2%)	53.52 (+2%)	84.45 (+1%)	58.98 (+1%)
		50	72.68 (+6%)	57.08 (+9%)	85.89 (+3%)	63.65 (+9%)
		75	**75.76 (+10%)**	**60.97 (+17%)**	**87.52 (+5%)**	**66.82 (+14%)**
	degC	25	72.33 (+5%)	56.65 (+8%)	85.73 (+3%)	62.15 (+6%)
		50	74.72 (+9%)	59.65 (+14%)	86.97 (+4%)	64.98 (+11%)
		75	**74.96 (+9%)**	**59.95 (+15%)**	**87.11 (+4%)**	**66.14 (+13%)**
	prC	25	72.85 (+6%)	57.29 (+10%)	85.96 (+3%)	63.62 (+9%)
		50	**75.16 (+10%)**	**60.21 (+15%)**	**87.23 (+5%)**	**65.94 (+13%)**
		75	73.8 (+8%)	58.48 (+12%)	86.48 (+4%)	64.78 (+11%)

10% in terms of F-Measure, NMI, Rand Index and JC respectively. In case of Attack dataset too, hashtags are important feature for determining topics as there is significant decrease in LDA performance after removing hashtags.

In case of Election dataset, the performance after removing hashtags is decreased by 26%, 75%, 15% and 32% in terms of F-Measure, NMI, Rand Index and JC respectively. The performance after removing mentions is decreased by 1%, 0%, and 2% in terms of F-Measure, Rand Index, JC and increased by 4% in terms of NMI. The performance after removing both hashtags and mentions is decreased by 27%, 71%, 14%, and 33% in terms of F-Measure, NMI, Rand Index and JC respectively. In case of Election dataset too, hashtags are important feature for determining topics as the LDA performance drop is maximum after removing hashtags.

4.4.5 Comparative results of LDA, Seeded-LDA, BTM and HP-LDA over tweets

Tables 4.6, 4.7 and 4.8 show the comparative performance of LDA, BTM, Seeded-LDA, HP-LDA over three datasets using clustering evaluation metrics: F-Measure, Rand-Index, Jaccard Coefficient(JC) and Normalized Mutual Information(NMI).

TABLE 4.7: Comparative results of LDA, BTM, Seeded-LDA and Hashtag Prioritized LDA(HP-LDA) over Attack dataset

Method	Prioritization approach	% of hashtags used	F-Measure(%)	JC (%)	Rand Index(%)	NMI (%)
LDA [2]	-	-	45.68	29.6	72.79	32.91
BTM [74]	-	-	**52.39 (+15%)**	**35.5 (+20%)**	**72.88 (0%)**	**39.03 (+19%)**
Seeded-LDA [72]	Manual hashtags	-	**57.8 (+27%)**	**40.65 (+37%)**	**78.61 (+8%)**	**44.07 (+34%)**
HP-LDA	Manual hashtags	-	**53.88 (+18%)**	**36.87 (+25%)**	**76.38 (+5%)**	**40.3 (+22%)**
	all_hashtags	100	**57.9 (+27%)**	**40.75 (+38%)**	**78.57 (+8%)**	**43.66 (+33%)**
	betC	25	55.1 (+21%)	38.03 (28%)	77.56 (+7%)	41.29 (+25%)
		50	55.37 (+21%)	38.28 (+29%)	77.36 (+6%)	40.14 (+22%)
		75	**56.48 (+24%)**	**39.35 (+33%)**	**77.81 (+7%)**	**43.22 (+31%)**
	closC	25	**59.39 (+30%)**	**42.24 (+43%)**	**79.18 (+9%)**	**45.59 (+39%)**
		50	53.42 (+17%)	36.44 (+23%)	76.34 (+5%)	39.84 (+21%)
		75	58.96 (+29%)	41.81 (+41%)	78.98 (+9%)	45.34 (+38%)
	degC	25	**62.67 (+37%)**	**45.64 (+54%)**	**80.92 (+11%)**	**49.97 (+52%)**
		50	49.26 (+8%)	32.68 (+10%)	74.36 (+2%)	35.36 (+7%)
		75	47.85 (+5%)	31.45 (+6%)	73.9 (+2%)	33.64 (+2%)
	prC	25	55.32 (+21%)	38.24 (+29%)	77.44 (+6%)	44.5 (+35%)
		50	50.92 (+11%)	34.16 (+15%)	75.18 (+3%)	37.36 (+14%)
		75	**58.25 (+28%)**	**41.1 (+39%)**	**78.28 (+8%)**	**43.85 (+33%)**

TABLE 4.8: Comparative results of LDA, BTM, Seeded-LDA and Hashtag Prioritized LDA(HP-LDA) over Election dataset.

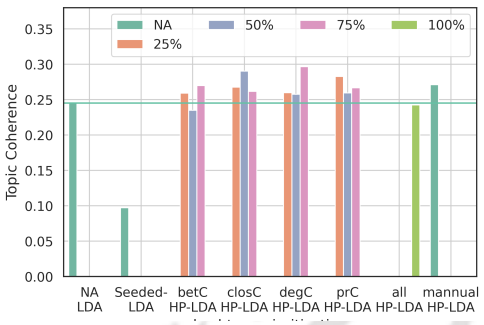
Method	Prioritization approach	% of hashtags used	F-Measure(%)	JC (%)	Rand Index(%)	NMI (%)
LDA [2]	-	-	49.39	32.8	65.11	22.96
BTM [74]	-	-	43.16 (-13%)	27.52 (-16%)	58.71 (-10%)	11.05 (-52%)
Seeded-LDA [72]	Manual hashtags	-	**69.95 (+42%)**	**53.79 (+64%)**	**79.16 (+22%)**	**47.99 (+109%)**
HP-LDA	Manual hashtags	-	48.35 (-2%)	31.88 (-3%)	63.78 (-2%)	23.42 (+2%)
	all_hashtags	-	51.75 (+5%)	34.91 (+6%)	66.87 (+3%)	27.14 (+18%)
	betC	25	49.57 (0%)	32.95 (0%)	64.69 (-1%)	25.14 (+9%)
		50	**57.64 (+11%)**	**40.49 (+16%)**	**69.79 (+4%)**	**34.91 (+29%)**
		75	54.2 (+10%)	37.17 (+13%)	68.06 (+5%)	28.53 (+24%)
	closC	25	73.7 (+49%)	58.35 (+78%)	81.71 (+25%)	52.33 (+128%)
		50	51.63 (+5%)	34.8 (+6%)	66.16 (+2%)	23.84 (+4%)
		75	**75.62 (+53%)**	**60.8 (+85%)**	**82.82 (+27%)**	**56.81 (+147%)**
	degC	25	**69.82 (+41%)**	**53.63 (+64%)**	**78.7 (+21%)**	**49.08 (+114%)**
		50	49.13 (-1%)	32.56 (-1%)	64.61 (-1%)	26.6 (+16%)
		75	66.1 (+34%)	49.37 (+51%)	76.05 (+17%)	44.53 (+94%)
	prC	25	61.56 (+25%)	44.46 (+36%)	73.25 (+13%)	38.01 (+66%)
		50	**72.36 (+47%)**	**56.69 (+73%)**	**80.91 (+24%)**	**53.99 (+135%)**
		75	63.27 (+28%)	46.28 (+41%)	73.7 (+13%)	43.19 (+88%)

LDA performance over Heterogeneous dataset, as given in Table 4.6, is 68.61%, 52.2%, 83.38%, and 58.46% in terms of F-Measure, JC, Rand-Index and NMI. All setups of HP-LDA over Heterogeneous dataset (except using top 50% of hashtags

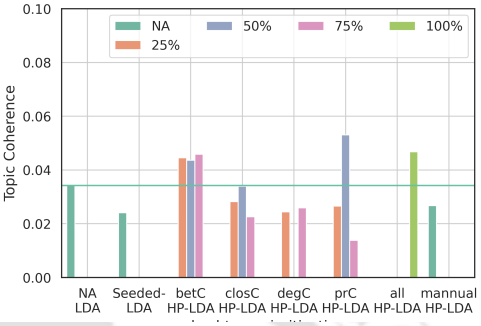
sorted by betweenness centrality score) perform better than LDA. The performance improvement in case of using all hashtags as prioritized word is 6%, 9%, 3%, and 9% more than LDA in terms of F-Measure, JC, Rand-Index and NMI. For closeness centrality and degree centrality, the clustering performance of HP-LDA increases as we increase the percentage of hashtags used as prioritized hashtags and the best performance is obtained with prioritization of top 75% of total hashtags ranked by respective centrality score. In case of betweenness centrality and PageRank centrality, the best performance of HP-LDA is obtained with prioritization of top 75% and 50% of total hashtags sorted by respective centrality scores. The overall best performance of HP-LDA is obtained at selecting 75% of hashtags using closeness centrality, which is 10%, 17%, 5%, and 14% more than LDA in terms of F-Measure, JC, Rand-Index and NMI. The performance of Seeded-LDA with manually selected hashtags for each class is 12%, 19%, 6% and 7% is more than LDA whereas performance of HP-LDA using same hashtags is 9%, 15%, 4%, and 13% more than LDA in terms of F-Measure, JC, Rand-Index and NMI. The performance of Seeded-LDA in case of Heterogeneous dataset is comparable with selecting 75% of total hashtags using closeness centrality and 50% of total hashtags using page rank centrality. The performance of BTM is 8%, 12% and 9% lesser than LDA in terms of F-Measure, JC, and Rand Index, whereas its performance in terms of NMI is 2% more than LDA.

LDA performance over Attack dataset, as given in Table 4.7, is 45.68%, 29.6%, 72.79%, and 32.91% in terms of F-Measure, JC, Rand-Index and NMI respectively. All setups of HP-LDA performs over Homogeneous dataset perform better than LDA. The performance improvement in case of using all hashtags as prioritized words is 27%, 38%, 8%, and 33% more than LDA in terms of F-Measure, JC, Rand-Index and NMI. For closeness centrality and degree centrality, the best clustering performance of HP-LDA is obtained with prioritization of top 25% of total hashtags sorted by respective centrality scores. For selecting prioritized hashtags using betweenness and page rank centrality scores, the best clustering performance is obtained with prioritization of 75% of total hashtags. The overall all best performance of HP-LDA is obtained with prioritization of top 25% of hashtags using degree centrality score, which is 37%, 54%, 11%, and 52% more than LDA in terms of F-Measure, JC, Rand-Index and NMI. Seeded-LDA with manually selected hashtags for each class as prioritized words is 27%, 37%, 8%, and 34% better than LDA whereas performance of HP-LDA using same hashtags as prioritized words is 18%, 25%, 5%, and 22% better than LDA in terms of F-Measure, JC, Rand-Index and NMI respectively. The performance of BTM is 15%, 20%,

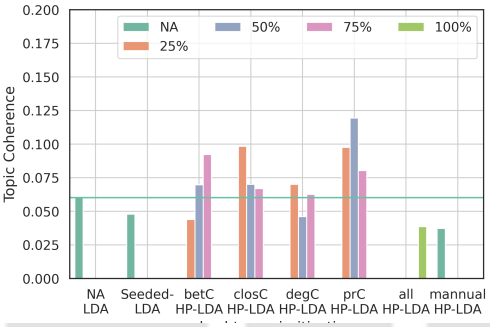
and 19% more than LDA in terms of F-Measure, JC, and NMI. The performance of HP-LDA with using all hashtags, top 25% of total hashtags in case of closeness centrality and degree centrality, and top 75% of hashtags in case of page rank centrality as prioritized words is greater than both Seeded-LDA with manually selected hashtags prioritized words and BTM.



(A) TC over Heterogeneous dataset.



(B) TC over Attack dataset.



(C) TC over Election dataset.

FIGURE 4.4: Comparison of topic coherence (TC) of the proposed HP-LDA with LDA and Seeded-LDA over Heterogeneous, Attack, and Election datasets.

LDA performance over Election dataset, as given in Table 4.8, is 49.31%, 32.8%, 65.11%, and 22.96% in terms of F-Measure, JC, Rand-Index and NMI respectively. All setups of HP-LDA (except manually selected hashtags and using top 50% of hashtags sorted by degree centrality score) perform better than LDA. The performance improvement in case of using all hashtags as prioritized word is 5%, 6%, 3%, and 18% more than LDA in terms of F-Measure, JC, Rand-Index and NMI. For closeness centrality and degree centrality, the best clustering performance of HP-LDA is obtained with prioritization of top 75% and top 25% of total hashtags sorted by respective centrality scores. For betweenness and page rank centrality approaches, the best clustering performance is obtained with prioritization of top 50% of total hashtags using respective centrality scores. The overall all best performance of HP-LDA is obtained with the prioritization of top 50% of hashtags

using page rank centrality score, which is 47%, 73%, 24%, and 135% more than LDA in terms of F-Measure, JC, Rand-Index and NMI. Seeded-LDA with manually selected hashtags for each class as prioritized words is 42%, 64%, 22%, and 100% better than LDA whereas performance of HP-LDA using same hashtags as prioritized words is 2%, 3%, 2%, less than LDA in terms of F-Measure, JC, Rand-Index and 2% more than LDA in terms of NMI respectively. The performance of BTM is 13%, 16%, 10% and 19% less than LDA in terms of F-Measure, JC, Rand Index and NMI. The performance of HP-LDA with using all hashtags, top 75% of total hashtags in case of closeness centrality, and top 50% of hashtags in case of page rank centrality as prioritized words is greater than both Seeded-LDA with manually selected hashtags prioritized word and BTM.

Further, Figure 4.4 presents comparative analysis of topic coherence [55, 56] obtained using LDA, Seeded-LDA and the proposed HP-LDA over the three datasets. From the figure, we observe that most of the hashtag prioritization approaches using word-to-word Co-occurrence centrality measures such as betweenness centrality (betC), closeness centrality measure (closC), degree centrality (degC), page rank centrality(prC), and using all hashtags performs superior to LDA and seeded-LDA. The topic coherence of seeded-LDA is slightly lesser than both HP-LDA and LDA over the three datasets. One critical observation is that for small overlapping dataset (Heterogeneous dataset), the majority of the methods (including baseline methods) provide reasonable performance. For the high overlapping datasets (Attack and Election datasets), the proposed method (HP-LDA) significantly outperform the baseline.

4.5 Prioritized Named Entities LDA (PNE-LDA)

In this section, we discuss our proposed prioritized named entity driven LDA *(PNE-LDA)*. PNE-LDA is similar to LDA, in terms of graphical plate diagram and its generative algorithm. Instead of prioritizing hashtags in tweets, PNE-LDA has considered named entities as prioritized words. For identifying the named entities, we have used Standford NER [138]. For the bomb blast dataset, since there is a need to identify Indian named entities, we have used and adapted Stanford NER, which is trained to recognize India named entities. Further, these named entities can be assigned different priority. However, for simplicity, this study assigns equal priority to all the named entities present in the documents. Once we identify representative named entities, the proposed PNE-LDA considers these entities

TABLE 4.9: Characteristics of the experimental datasets used for PNE-LDA over news media. NE represent Named Entity

Dataset	#Doc	#class	Avg. #NE per doc	Avg. #doc per class
Bomb Blast	855	53	225	16
Reuters-21578-R	5485	8	70	686
20-Newsgroups	11293	20	152	565

as prioritized keywords and rest as the non-prioritized keywords. Furthermore, this section presents the characteristics of news media datasets considered for experiment, experimental setups along with results of PNE-LDA model and its counterparts.

4.5.1 Datasets used for PNE-LDA over news media

For evaluating PNE-LDA over news media considering named entities as prioritized word, we have experimented with three datasets namely Bomb Blast, Reuters-21578-R, and 20-Newsgroups respectively as described in Table 4.9. The Bomb Blast dataset is our locally collected and processed dataset, which consist of 855 news articles reporting 53 different bomb blast events occurred in different parts of India. The Reuters-21578-R dataset consists of 5,485 documents spanning across 8 clusters and 20-Newsgroups dataset consists of 11,293 documents spanning across 20 clusters. Among the three datasets, Bomb Blast has the most occurrences of named entities, while Reuters dataset has the least occurrences of named entities in the documents. Bomb Blast dataset mostly has the person, location and organization name, whereas 20-Newsgroups dataset has only person and organization names. Reuters-21578-R dataset is mostly related to business articles and hence has a limited number of named entities.

4.5.2 Experimental set up for PNE-LDA and its counterparts

Similar to HP-LDA, we removed stopwords and punctuations using NLTK library from the title and content of the news articles. Following are the different experimental setups:

TABLE 4.10: Evaluation of LDA, Seeded-LDA, PNE-LDA (proposed) in terms of F-measure and Rand Index over Bomb Blast, Reuters-21578 and 20-Newsgroups datasets.

Model	Bomb Blast		Reuters-21578		20-Newsgroups	
	F-Measure (%)	Rand Index(%)	F-Measure (%)	Rand Index(%)	F-Measure (%)	Rand Index(%)
LDA	8.02	79.87	60.52	78.00	13.82	60.56
Seeded-LDA	8.68 (+8%)	83.19 (+4%)	**62.34** (+3%)	**78.55** (+1%)	33.56 (+143%)	91.26 (+51%)
PNE-LDA	**8.87** (+11%)	**86.72** (+9%)	53.53 (-12%)	76.03 (-3%)	**50.03** (+262%)	**94.15** (+55%)

- **LDA**: We set document-topic Dirichlet parameter $\alpha = 0.1$ for all the three datasets. We set topic-word Dirichlet parameter η to 0.2, 0.3, and 0.2 for Bomb Blast, Reuters-21578-R and 20-Newsgroups datasets respectively. We have chosen the particular value of α, and η after finding it empirically better than other value in the set of { 0.1, 0.2, 0.3, 0.4}.
- **PNE-LDA**: We have set value of $\alpha = 0.1$ same as LDA setups. We set the value of topic-prioritized named entity Dirichlet parameter η_1 as 0.1 and topic-general word Dirichlet parameter η_2 as 0.2 for Bomb Blast dataset. For Reuters-21578-R dataset, we have set η_1 as 0.2 and η_2 as 0.3 respectively. For 20-Newsgroup, we have set η_1 as 0.1 and η_2 as 0.2 respectively. Although we have experimented with other combination of η_1 in {0.1, 0.2, 0.3, 0.4} and η_2 in {0.1, 0.2, 0.3, 0.4}, but we are reporting parameters with the best performance with respect to both F-Measure and Rand Index.
- **Seeded-LDA**: For Seeded-LDA we have used the same value of α, η_1 and η_2 as PNE-LDA corresponding to three datasets.

4.5.3 Comparative results of LDA, Seeded-LDA, and PNE-LDA over news media

To investigate the performance of different methods, we have used the document clustering task to measure the quality of topics given by LDA, Seeded-LDA and PNE-LDA. For all the experimental setups, we consider the number of topics as the number of clusters present in the respective datasets, as shown in Table 4.9. LDA, Seeded-LDA and PNE-LDA returns the document-topic proportions. Each document is assigned to the cluster defined by the topic with maximum proportions. Once each document is assigned with a cluster ID (which is topic ID with maximum proportions), we evaluate clustering performance using F-Measure and Rand Index.

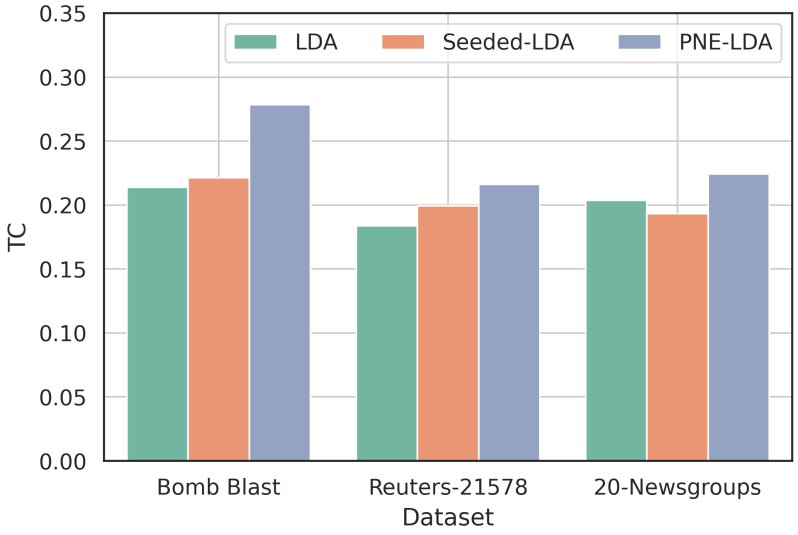


FIGURE 4.5: Comparison of topic quality in terms of topic coherence *(TC)* using LDA, Seeded-LDA, and PNE-LDA (proposed) over Bomb Blast, Reuters-21578, and 20-Newsgroups datasets.

Observations : Table 4.10 presents comparative performance of the three topic models namely LDA, Seeded-LDA and PNE-LDA (proposed) in terms of F-measure, and Rand Index over three datasets Bomb Blast, Reuters-21570, and 20-Newsgroups respectively. The brief details about the evaluation of topic models using clustering metrics is included in the section 2.3. In case of Bomb Blast dataset, the percentage of improvement of PNE-LDA over LDA is 11% and 9% in terms of F-measure and Rand-Index respectively. Similarly, in case of 20-Newsgroup dataset, the percentage of improvement is 143% and 55% respectively in terms of F-measure and Rand Index respectively. As mentioned in the subsection 4.5.1, both LDA and seeded LDA for both Bomb Blast and 20-Newsgroups datasets, which may because of large improvement in the performance of PNE-LDA over LDA, and Seeded-LDA in case of the two datasets. Whereas, in less named entity-driven dataset (Reuters-21578-R), PNE-LDA under-performs both LDA and Seeded-LDA. The low value of F-measure in case of Bomb blast dataset can be attributed to large number of classes (53) and lesser average number of documents per class.

Further, Figure 4.5 compares topic quality of three models (LDA, Seeded-LDA and PNE-LDA) in terms of topic coherence [55, 56] over the three datasets. From the figure, it is evident that the topics given by PNE-LDA is more coherent and meaningful as compared to LDA and Seeded-LDA. It is evident from above observation that the proposed PNE-LDA is more effective to determine real-world events which can be represented by named entities defining the topics. Moreover,

the proposed PNE-LDA model shows improvement of 11% and 30% in terms of F-measure and topic coherence even with the dataset with low document support (16) per class.

4.6 Summary

This chapter demonstrates the importance of hashtags for the topic modeling in tweets. The proposed Hashtag Prioritized LDA *(HP-LDA)* outperforms LDA, and Biterm Topic Model using different hashtags selection approach over three datasets. The HP-LDA outperforms Seeded-LDA in the case of Election and Attack dataset and has comparable performance in the Heterogeneous dataset. Further, this chapter extends HP-LDA as Prioritized Named Entity driven LDA *(PNE-LDA)* for news media by considering named entities as prioritized words. Experimental results over three datasets show that PNE-LDA outperforms LDA and Seeded-LDA in entity-driven datasets. In our future exploration, would like to compare the performance of proposed Hashtag-prioritized LDA (HP-LDA) with the state-of-the-art supervised classifier.

Chapter 5

Downstream Application of LDA – A case study on terror attack prediction

In the previous chapters, the methods for improving topic modeling performance in noisy and under-specified scenarios have been discussed. In this chapter, we further present a downstream application of LDA in relation prediction, i.e., predicting future relations between attributes of terror attacks in particular. In the past, LDA has been used explored in relation mining. Author Topic *(AT)* [20] and Author Recipient Topic *(ART)* [21] are few of such studies. Unlike these models, we focus on incorporating topics discovered using LDA from external sources in enhancing relation prediction. However, this study explores LDA over well form documents, i.e., news articles, instead of noisy tweets, and applied it for link prediction over a network constructed from the attributes of terror attacks. Application of our proposed models over tweets for predicting the relationships between tweet related attributes such as hashtags-to-hashtags, mention-to-mentions, hashtags-to-mention, hashtags-to-users, etc., is not included and left as a part of our future works.

5.1 Introduction

With the increase in availability of digitized data related to terrorist activities, Social Network Analysis *(SNA)* has garnered increasing attention in analyzing counter-terrorism related data in recent time. Several works are presented in the

literature to understand terrorist network using various SNA methods such as link prediction [139], structural analysis [140, 141, 142], modeling [143, 144] etc. Major concerns in all these studies are (i) the size of the datasets and (ii) the nature of the datasets. Majority of the datasets used in above studies are small, and the underlying networks are homogeneous in nature (nodes are of same type, i.e., nodes of the network are mostly either terrorists or terrorist organizations). Getting a large terrorist network of high quality is one of the core challenges in academic research because of several reasons as described in [140]; *size, incompleteness, fuzzy boundary* and *dynamics*. Recently, various agencies have made efforts to create large databases related to terrorist activities in the public domain. Some of such datasets are the *Global Terror Data* (GTD) [1], *South Asian Terror Portal* (SATP) [2] etc. However, these databases provide information in semi-structured or unstructured forms. Constructing a homogeneous network out of such dataset is a non-trivial task. Considering the nature of such datasets, it is important to explore the methods which can deal with the semi-structured or unstructured nature of the datasets.

An event of a terrorist attack is often defined by a set of attributes such as terrorist organizations involved, accused terrorists, place of attack, target type, materials used, victims etc. Analysis related to counter-terrorism often needs to study the relationship between different attributes such as *terrorist group and target type*, or *material used and target type* or *country and potential terrorist organizations* etc. Since relationships between any two attributes are often influenced by other attributes, it is important to consider other attributes while exploring hidden relationships between them. Motivated by the above two factors - *the unstructured form of data and different set of attributes*, the objective of this chapter is to predict links (hidden or future) between different attributes of the event of terrorist attacks by considering the topic discovered from the news articles. This chapter investigates the efficacy of latent topics obtained by LDA [2] in predicting relationships between different types of nodes over GTD dataset, a large semi-structure dataset reporting different terrorist attacks over four decades. From various experimental analysis over GTD dataset, it is evident that LDA provides promising performances in relation prediction over different attributes of a terrorist network.

[1] http://www.start.umd.edu/gtd/
[2] http://www.satp.org/

5.1.1 Contribution

The major contributions of this chapter are:

- Proposed an LDA based approach to predict the link between different attributes of a terrorist attack over Global Terror Data (GTD).
- Evaluated the importance of latent topic information in predicting future relationship between attributes of terrorist attack as compared to popular SNA methods such as Common Neighbor, Jaccard Index, and Resource Allocation.

5.2 Related Studies

LDA has been adopted to mine the relationship between entities and topics from research publications and emails datasets. Rosen et al. [20] proposed Author Topic *(AT)* to mine the relationship between author and research topics. Similar to LDA, AT models every topic as a Multinomial distribution over words in the vocabulary. LDA models every document as a Multinomial distribution over topics, whereas AT models every author as a Multinomial distribution of topics. Given an abstract of the paper, and a list of authors plus their past collaborators, AT generates the list of authors working in similar areas using the distance between authors via author-topic distribution. The distance between author i and authors j can be defined using symmetric KL divergence between topics distributions conditioned on each author as follows:

$$SKL(i,j) = \sum_{t=1}^{T} [\theta_{it} log(\frac{\theta_{it}}{\theta + jt}) + \theta_{jt} log(\frac{\theta_{jt}}{\theta_{it}})]$$

where T denotes the total number of topics, θ_{it} denotes the probability of author i writing about topic t and θ_{jt} denotes the probability of author j writing about topic t .

Tu et al. [57] proposed Citation Author Topic *(CAT)* extending AT, to model the relationship between author and research topics by including citation information. CAT adds one more layer of citation-topic Multinomial apart from author-topic and topic-word Multinomial present in AT model. The generative process of CAT model samples an author from the observed Multinomial distribution, thereafter

samples a topic conditioned on the author, and finally generates a word and cited author conditioned. CAT model is used to build an *expert search* system, which returns a list of researchers corresponding to the input query words. CAT ranks every author (a) corresponding to the query words (W) as follows:

$$P(W,a) = \sum_{w_i \in W} \alpha_i \sum_{t=1}^{T} P(w_i, a|t, c_a) P(t, c_a)$$

$$= \sum_{w_i \in W} \alpha_i \sum_{t=1}^{T} P(w_i|t) P(a|t) P(c_a|t) P(t)$$

where α_i is the inverse document frequency for the word w_i, T is the total number of topics, c_a represents that the author is one of the cited authors in the corpus.

McCallum et al. [21] proposed an Author Recipient Topic *(ART)* model extending AT to mine links between peoples (sender and receiver of an email) over Enron and academic email datasets. Unlike traditional Social Network Analysis *(SNA)* methods, which only use link information, ART incorporates the textual content and topics to model the relationship between entities. The ART model can predict people roles and calculate similarities between people based on the role and topics. ART models the topic distribution of a document conditioned on author and individual recipients. ART measures the equivalence of role between two persons using inverse Jensen-Shannon divergence, assuming that two people with similar roles will have a similar probability distribution over their communication partners.

In the above studies, the relations between topic and entities or relations between same type of entities are explored using an extension of LDA. This chapter targets the problem of link prediction between a pair of attributes of a terror attack without modifying the basic LDA algorithm.

5.3 Datasets

The dataset *Global Terror Data (GTD)* used in this work is collected from *National Consortium for the Study of Terrorism and Responses to Terrorism*, University of Maryland [145]. This repository contains information about more than 140,000 terrorist attacks collected globally over the years 1970 to 2014. Samples of GTD datasets is shown in the Table 5.1. It consists of the attributes such as victim location (Country, Region, Province, City), modus-operandi of terrorist groups

TABLE 5.1: Sample of GTD used for constructing heterogeneous terrorist attack network.

ID	Country	Region	Province	City	Attack type	Target type	Target subtype	Group name	Weapon type	Weapon subtype
1	India	South Asia	Assam	Dibrugarh	Bombing	Educational Institution	School/ University	ULFA	Explosives	Grenade
2	India	South Asia	Orissa	Jajpur	Bombing	Transportation	Bridge/ Car Tunnel	CPI-Maoist	Explosive	Land Mine
3	India	South Asia	Assam	Kokrajhar	Facility	Transportation	Train/ Train Tracks	NDFB	Sabotage	Equipment
4	India	South Asia	J&K	Bijbehara	Bombing	Police	Police Patrol	LeT	Explosives	Vehicle
5	Nigeria	Sub-Saharan Africa	Borno	Maiduguri	Facility	Religious Figures	Place of Worship	Boko Haram	Incendiary	Arson/ Fire
6	Afghanistan	South Asia	Kandahar	Kandahar	Bombing	Religious Figures	Place of Worship	Taliban	Explosives	Suicide

(Attack type, Target type, Weapon type), name of the terrorist groups (Group name) etc.

Graph Generation: To evaluate the performance of the proposed method, we have considered various local proximity-based link prediction methods on a complex network. To execute these methods and the future relations (evaluation set), we construct the following network. We extract the following ten important features; Country, Region, Province, City, Attack type, Target type, Target subtype, Group name, Weapon type, and Weapon subtype corresponding to each terror attack. All the entries corresponding to terrorist group's name as "unknown/Unknown" are deleted from the dataset. We generate a graph by drawing an edge between these features if they are associated with the same event/attack. We partition the dataset into training and testing dataset as follows: i) Train dataset consists of all the terrorist events that happened between 1970 and 2009, and ii) Test dataset consists of all the terrorist events between 2010 and 2014. The training bipartite network for our proposed method is also constructed using the terror attacks associated with the above training set. Table 5.2 presents the number of nodes and edges in train and test network used by local similarity based SNA methods such as Common Neighbor *(CN)*, Jaccard Coefficient *(JC)*, Adamic Adar *(AA)*, Resource Allocation *(RA)*. The train dataset consists of 17326 nodes distributed over ten different types of nodes and 239193 edges distributed over forty-five different types of edges. Table 5.3 presents the distribution of different nodes types in the Train dataset. Similarly, the Test dataset consists of 1783 nodes distributed over ten different types of nodes and 38499 edges distributed over forty-five different type of edges.

Preparing text document corpus for Topic modeling: Each row of the GTD dataset is associated with links for news articles reporting about the terrorist attack. We crawled the news articles present in the GTD dataset. For each terrorist

TABLE 5.2: Network characteristics used local similarity based SNA methods Common Neighbour (CN), Jaccard Coefficient (JC), Adamic Adar (AA), Resource Allocation (RA)

# of Node (Training)	# of Edges (Training)	# of Nodes (Testing)	# of Edges (Testing)
17326	239193	1783	38499

TABLE 5.3: Number of different type of Nodes in Train dataset.

S.No	Type of Node	# of Node count	S.No	Type of Node	# of Node count
1	Country	186	6	Target type	22
2	Group	2562	7	Target subtype	111
3	City	13034	8	Weapon type	12
4	Region	12	9	Weapon subtype	28
5	Province	1350	10	Attack type	9

attack, we made a text document by appending the content of corresponding news articles and the selected ten features. We considered the text documents corresponding to terrorist attacks from 1970 to 2009 as the Train dataset for LDA. Train dataset for LDA consists of 1,41,966 documents with 35,021 unique words.

5.4 Methodology

Using *(LDA)*, we first discover K topic, i.e., $\mathcal{T} = \{t_1, t_2, ..., t_k\}$ from document collection $\mathcal{D} = \{d_1, d_2, d_3, ..., d_m\}$. Our goal is to predict the relationship between attribute pairs of GTD through latent topics obtained using LDA. Given a set of attribute values extracted from GTD (denoted by $\mathcal{W} = \{w_1, w_2, w_3, ..., w_n)$, we construct a bipartite graph between the elements of $\mathcal{T}$ and, $\mathcal{W}$ as shown in Figure 5.1. An edge in the bipartite graph represents an association of topic and an attribute value. The weight of the edges are the word distribution in the topic. If w_i and w_j denote two attribute values in $\mathcal{W}$, the similarity score between w_i and w_j is defined as follows.

$$score(w_i, w_j) = \sum_{k=0}^{k=K-1} \begin{cases} \left\{ \frac{Pr(w_i,t_k)}{log(r(w_i,t_k))} + \frac{Pr(w_j,t_k)}{log(r(w_j,t_k))} \right\} & \text{if } score(w_i, t_k) > \theta \\ & \text{and } score(w_j, t_k) > \theta \\ 0 & \text{Otherwise} \end{cases} \quad (5.1)$$

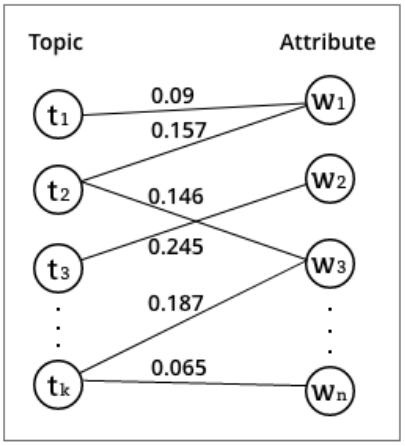


FIGURE 5.1: Bipartite graph between Attribute and Topic obtained from LDA

ALGORITHM 5: Link Prediction Score Between two Attributes

Input: A:Document Word matrix, K: number of topics, L: attribute pairs list

Output: R:Score of each attribute pairs in L

Initialize LDA hyper parameters;
$R = [\,]$;
`// Get topic-word and document-topic distribution using LDA`
topic-word, doc-topic = LDA(A, K) ;
Create a bipartite graph between topic and attributes using topic-word as shown in figure 5.1 ;
`// Calculate scores between pairs of attributes using equation 5.1`
for $(w_i, w_j) \in L$ **do**
    $s(w_i, w_j) = score(w_i, w_j)$;
    $R[< w_i, w_j >] = s(w_i, w_j)$;
end
Return R ;

where θ is a user-defined threshold, $Pr(w_i, t_k)$ represents probability of w_i in t_k topic, and $r(w_i, t_k)$ represents the rank of w_i in topic k defined by probability of words in the topic. This similarity score is considered as the possible likelihood of having a relation between two attributes in the future. The proposed relation prediction between two attributes is summarized in the Algorithm 5.

5.5 Results and Discussion

5.5.1 Experimental setup

The following baseline methods are considered:

- **Common Neighbor** *(CN)*: The common neighbor score between two nodes x and y in a network is defined by the number of common nodes directly incident to the nodes x and y i.e., $score_{CN}(\text{x, y}) = |n(x) \cap n(y)|$, where $n(x)$ and $n(y)$ denote the neighbor nodes of x and y in the network.
- **Jaccard Coefficient** *(JC)*: The Jaccard Coefficient score between two nodes x and y in a network is defined by $score_{JC}(\text{x, y}) = \frac{|n(x) \cap n(y)|}{|n(x) \cup n(y)|}$.
- **Adamic Adar** *(AA)*: Traditional AA index between two nodes x and y for a network is defined as $score_{AA}(x, y) = \sum_{z \in n(x) \cap n(y)} \frac{1}{log(|n(z)|)}$.
- **Resource Allocation** *(RA)*: RA index between two nodes x and y for a network is defined as $score_{RA}(x, y) = \sum_{z \in n(x) \cap n(y)} \frac{1}{|n(z)|}$.

Parameter setup for proposed method (LDA): The relation score between two attributes of a terror attack is calculated using the algorithm 5. We set the threshold to 0.0001 (this threshold has been empirically chosen). We have experimented with the above algorithm with different values of K (number of topics) ranging from 5 to 50 with a gap of 5. Empirically, we found that topic 45 gives the best result for almost all cases. We are reporting only the peak results in Table 5.5.

5.5.1.1 Evaluation

We evaluate performance of all the link prediction models by finding their Area under ROC curve *(AUC)* score. For this purpose, we generate 500000 edges randomly which are not existing in test graph, which is also called as *non-existing edges* (n_{ne}) or *negative edges*. The set of edges that are already present in the test graph are called as *existing edges* (n_e) or *positive edges*. AUC score for evaluating the performance of link prediction is given by following formula [146]:

$$AUC = \frac{n_1 + 0.5n_2}{n_{ne} * n_e} \tag{5.2}$$

TABLE 5.4: Different type of Test Edges considered for Evaluating Link Predictions.

Category \ Edge type	Cn-gp	Cty-gp	Tar-gp	Weap-gp
All Edges	350	1020	761	413
Missing Edges	104	374	200	71

where n_e = number of existing test edges (positive edges), n_{ne} = number of non-existing test edges (negative edges), n_1 = number of times link prediction score for existing test edge (positive edges) is greater than other non-existing test edges (negative edges), n_2 = number of times link prediction score for existing test edge is equal to other non-existing test edges.

We assess the performance of link predictors on two different sets of test edges. For the first type of test edges (i.e., All Edges), we consider all the edges appeared between years 2010 to 2014. For the second type of test edges (i.e., Missing Edges – the edges that were not present in the training edges), we consider only new edges appeared between 2010 and 2014. The motivation for dividing test data into these two forms is, we want to evaluate the models in terms of:

- Performance on seen as well as unseen connectivity (All Edges) – total test edges. Previously observed relations may continue to exist. For example, the same terrorist group may attack in the same location again in the future.
- Performance on unseen connectivity (Missing Edges) – the set of test edges that were not present in training dataset. For example, a terrorist group may attack in a new location where they have not attacked before.

In this particular study, we have considered four pairs of attributes relations; Country of attack vs Terrorist Group (**Cn-gp**), City of attack vs terrorist group (**Cty-gp**), Target type vs terrorist group (**Tar-gp**), and Weapon type vs terrorist group (**Weap-gp**). Table 5.4 presents count of different type of test edges considered under *All Edges* and *Missing Edges*.

5.5.2 Experimental Observation

Table 5.5 presents the Average AUC score for all four types of relationships namely: i) *Cn-gp*, ii) *Cty-gp*, iii) *Tar-gp*, and iv) *Weap-gp* for *All Edges* and

TABLE 5.5: Comparison of average AUC score for Link Prediction on All edges and missing edges using Common Neighbour (CN), Jaccard Coefficient (JC), Adamic Adar (AA), Resource Allocation (RA), and proposed LDA based approaches.

	All Edges					Missing Edges				
	CN	**JC**	**AA**	**RA**	**Proposed approach**	**CN**	**JC**	**AA**	**RA**	**Proposed approach**
Cn-gp	0.78	0.56	0.78	0.56	**0.82**	0.67	0.54	0.68	0.54	**0.71**
Cty-gp	0.89	0.42	**0.90**	0.42	0.85	**0.82**	0.45	**0.82**	0.45	0.77
Tar-gp	0.74	0.47	**0.75**	0.47	0.68	0.53	0.37	0.53	0.37	**0.57**
Weap-gp	**0.82**	0.33	**0.82**	0.33	0.70	0.62	0.30	0.62	0.30	**0.64**

Missing Edges using local similarity based SNA methods (CN, JC, AA, RA) and our proposed LDA based approach. In case of *All Edges*, all the four types of relations are predicted with a convincing AUC score for *All Edges* using LDA, and local similarity based SNA approaches. LDA based approach outperforms local similarity based SNA approaches in predicting Cn-gp relation. AA of SNA based approach outperforms LDA and other SNA based approaches (CN, JC, RA) in predicting Cty-gp, Tar-gp, and Weap-gp relations. Average AUC score of LDA based approach in predicting Cty-gp, Tar-gp, and Weap-gp relation outperforms JC and RA based approaches.

In the case of *Missing Edges*, LDA based approach outperforms local similarity based SNA approaches (CN, JC, AA, RA) in predicting Cn-gp, Tar-gp, and Weap-gp relations. In case of Cty-gp relation, CN and AA based approaches outperforms LDA, JC, RA based approaches in terms of average AUC score. Average AUC score of LDA based approach in predicting Cn-gp and Cty-gp relations is more than 0.70, whereas Tar-gp and Weap-gp achieve a relatively lower average AUC score. Further, the AUC score for predicting *Missing Edges* by all approaches is lower compared to *All Edges* for all the four types of relations. This observation may be attributed to the fact that majorities of terrorist attacks are repetitive in nature, with a similar history of targets and weapons used in attacks. In case of both *All Edges* and *Missing Edges*, LDA based approach performs better in predicting Cn-gp and Cty-gp relations as compared to Tar-gp and Weap-gp relations.

5.6 Summary

This chapter explored the topic modeling for predicting links in the future over Global Terror Data (GTD). It is perceptible from the experimental results that

LDA based methods are efficient for predicting the link between different attributes of a terrorist network. In this chapter, we applied our LDA based approach of link prediction using documents from news articles reporting terrorist attacks. In future research exploration, we would like to evaluate different scoring function (such as multiplication in place of addition) in the equation 5.1 empirically for relation prediction over different attributes. Further, we plan to extend our proposed models over tweets for predicting relationship between tweet related attributes such as hashtags-to-hashtags, mention-to-mentions, hashtags-to-mention, hashtags-to-users. Moreover, we plan to incorporate different weights for different attributes such as hashtags, users, mentions in the LDA based link prediction approach over a tweet network.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, we studied the efficacy of hashtags in improving topic modeling performance over tweets. We explored different ways to utilize hashtags in solving the challenges faced by LDA of data sparsity and under specificity of tweets. Further, we explored the LDA for link prediction between key attributes of a terrorist attack.

Chapter 2 briefly discusses a few topic models such as LSA, NMF, pLSA, and LDA. We also discuss the different variants of LDA used in regular and short text and the significance of hashtags in short text for various application such as topic detection, event detection, sentiment analysis and personalized news recommendation.

In Chapter 3, we explored text-based and graph-based approaches of tweets expansions with semantically related hashtags. We present the significance of hashtags in LDA performance over tweets by experimenting with different combination of features such as i) *keywords* only, ii) *keywords with hashtags*, iii) *keywords with user mentions*, *keywords with hashtags and user mentions* over two datasets of distinct nature: Homogeneous dataset (classes with overlapping keywords and hashtags) and Heterogeneous datasets (classes with less overlapping of keywords and hashtags). In text-based approach, we explored BiLSTM and BERT-based methods to expand tweet with semantically related hashtags. And, in the case of graph-based approach for tweet expansion with semantically related hashtags, we explored 1-hop nearest neighbor and Graph Convolution Network *(GCN)* to model tweet representation using word co-occurrence graph. We evaluated the efficacy of

proposed tweet expansion by comparing the performance of LDA over expanded tweets compared to raw tweets. LDA performance after expanding tweets with the proposed expansion approaches improves significantly compared to raw tweet and hashtag pooling based tweets expansion. The results show that the percentage of improvement after tweet expansion in the Homogeneous dataset compared to raw tweet is more than the Heterogeneous dataset. Further, the proposed tweet expansion methods also perform better in finding distinct topic representation of classes with less document support.

Chapter 4 proposed prioritizing a few important keywords or tokens such as hashtags to guide the LDA in discovering a better topic over short and under-specified tweets. We proposed the Hashtag Prioritized LDA *(HP-LDA)* to prioritize the hashtags over other words in LDA over tweets and different approaches to select prioritized hashtags. The proposed HP-LDA outperforms LDA and Bi-Term Topic Model using different hashtags selection approaches over three datasets. The HP-LDA outperforms seeded-LDA in the Election and Attack dataset and has comparable performance in the Heterogeneous dataset. Further, this chapter extends HP-LDA as Prioritized Named Entity driven LDA *(PNE-LDA)* for news media by considering named entities as prioritized words. Experimental results over three datasets show that PNE-LDA outperforms LDA and seeded-LDA in entity-driven datasets.

In Chapter 5, we explored the efficacy of topic modeling for predicting links in future on terrorist network, namely, Global Terrorist Data (GTD). It is perceptible from the experimental results that LDA based methods are quite efficient for predicting future links between different attributes of a terrorist network. LDA based approach gives a decent AUC average score in predicting the future links between different attributes such as Country of attack vs Terrorist Group (**Cn-gp**), City of attack vs terrorist group (**Cty-gp**), Target type vs terrorist group (**Tar-gp**), and Weapon type vs terrorist group (**Weap-gp**).

6.2 Limitations and Future Works

This section discusses the limitations associated with the current study and some potential directions to explore in the future. A few of the major research directions for future explorations of the thesis work are as follows:

- **Incorporating temporal information in hashtag-based tweet expansion**: The creation time of tweet plays a crucial role in relating hashtags to the topic of the tweets. Considering temporal information with text-based and graph-based models may improve predicting semantically related hashtags and thus improve the topic modeling performance over expanded tweets.

- **Considering distributed representation of words in Hashtag Prioritized LDA (HP-LDA)**: Most of the topic models consider words as a discrete variable using bag-of-word representation. Distributed representation of the words such as word2vec [147], fasText [148] have shown promising results in improving diverse natural language processing tasks. Recent topic modeling methods such Gaussian LDA [149], Embedded Topic Model *(ETM)* [56], and tBERT [134] have incorporated the distributed representation of words into LDA. We plan to incorporate the distributed word representation into HP-LDA.

- **Exploration of different topic models over hashtags-based expanded tweets**: Different topic models such as Biterm Topic Model (BTM) [41], Embedding-based Topic Model (ETM) [133], Topic modeling in embedding spaces [56], and tBert [134] have been proposed recently to improve the topic modeling performance over short-text. We plan to explore the effect of different topics models over hashtag-based expanded tweets.

- **Exploration of response of different supervised topic models and deep-learning based classifier over hashtags-based expanded tweets**: Different supervised topic models such as Labeled LDA [64] have been proposed to incorporate the label information into the topic discovery process. We would like to study the performance of supervised topic models and supervised deep learning-based classifier [135] over the hashtag-based expanded tweets.

- **Comparison of state-of-the art deep learning based supervised classifier with the proposed Hashtag Prioritized LDA (HP-LDA**: In the Chapter-4, we have compared the performance of proposed Hashtag Prioritized LDA (HP-LDA) with unsupervised topic models only (such as LDA, Bigram-based topic models and Seeded-LDA). In the future, would like to compare the performance of proposed HP-LDA with state-of-the-art supervised classifier.

- **Predicting relation between different word types in tweets**: Relation prediction between different attributes of a terrorist attack has been

studied by incorporating latent topics discovered from the news publications. However, the influence of latent topics (discovered using the proposed enhanced LDA over tweet collection) in predicting various relationships in tweets such as hashtags-to-hashtags, mention-to-mentions, hashtags-to-mention, hashtags-to-users, etc. has not been included in the thesis. We plan to incorporate the enhanced LDA over tweets to predict the relationship between different attributes in future work. Further, we would like to evaluate different scoring function (such as multiplication in place of addition) in the equation 5.1 empirically for relation prediction over different attributes.

Bibliography

[1] T. L. Griffiths and M. Steyvers, "Finding scientific topics," *Proceedings of the National Academy of Sciences*, vol. 101, no. suppl 1, pp. 5228–5235, April 2004.

[2] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, pp. 993–1022, January 2003.

[3] X. Ni, J.-T. Sun, J. Hu, and Z. Chen, "Mining multilingual topics from wikipedia," in *Proceedings of the 18th International Conference on World Wide Web*, ser. WWW '09. New York, NY, United States: Association for Computing Machinery, April 2009, pp. 1155–1156.

[4] M. Pavlinek and V. Podgorelec, "Text classification method based on self-training and lda topic models," *Expert Systems with Applications*, vol. 80, pp. 83–93, September 2017.

[5] J. Jedrzejowicz and M. Zakrzewska, "Text classification using lda-w2v hybrid algorithm," in *Intelligent Decision Technologies 2019*, ser. Smart Innovation, Systems and Technologies, J. L. Czarnowski I., Howlett R., Ed., vol. 142. Singapore: Springer, July 2020, pp. 227–237.

[6] M. Lienou, H. Maitre, and M. Datcu, "Semantic annotation of satellite images using latent dirichlet allocation," *IEEE Geoscience and Remote Sensing Letters*, vol. 7, no. 1, pp. 28–32, January 2010.

[7] Q. Guo, N. Li, Y. Yang, and G. Wu, "Supervised lda for image annotation," in *2011 IEEE International Conference on Systems, Man, and Cybernetics*, Anchorage, Alaska, USA, October 2011, pp. 471–476.

[8] C.-T. Nguyen, D.-C. Zhan, and Z.-H. Zhou, "Multi-modal image annotation with multi-instance multi-label lda," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, ser. IJCAI '13, Beijing, China, August 2013, pp. 1558–1564.

[9] D. Hu and L. K. Saul, "A probabilistic topic model for unsupervised learning of musical key-profiles." in *10th International Society for Music Information Retrieval Conference*, ser. ISMIR '09, Kobe,Japan, October 2009, pp. 441–446.

[10] P. Huang, M. Wilson, D. Mayfield-Jones, V. Coneva, M. Frank, and D. H. Chitwood, "The evolution of western tonality: a corpus analysis of 24,000 songs from 190 composers over six centuries," *SocArXiv:10.31235/osf.io/btshk*, December 2017.

[11] C. Li, C. Yang, and Q. Jiang, "The research on text clustering based on lda joint model," *Journal of Intelligent & Fuzzy Systems*, vol. 32, no. 5, pp. 3655–3667, April 2017.

[12] X. Wei and W. B. Croft, "Lda-based document models for ad-hoc retrieval," in *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '06, Seattle Washington, USA, August 2006, pp. 178–185.

[13] D. M. Blei, "Probabilistic topic models," *Communications of the ACM*, vol. 55, no. 4, pp. 77–84, April 2012.

[14] T. A. Rana, Y.-N. Cheah, and S. Letchmunan, "Topic modeling in sentiment analysis: A systematic review," *Journal of ICT Research & Applications*, vol. 10, no. 1, pp. 76–93, February 2016.

[15] R. Hossain, M. R. K. R. Sarker, M. Mimo, A. Al Marouf, and B. Pandey, "Recommendation approach of english songs title based on latent dirichlet allocation applied on lyrics," in *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, Ras Al Khaimah, UAE, February 2019, pp. 1–4.

[16] N. Akhtar, M. S. Beg, and H. Javed, "Textrank enhanced topic model for query focussed text summarization," in *2019 IEEE 12th International Conference on Contemporary Computing (IC3)*, Noida, India, August 2019, pp. 1–6.

[17] Y. Wu, Y. Ding, X. Wang, and J. Xu, "Topic based automatic news recommendation using topic model and affinity propagation," in *2010 IEEE International Conference on Machine Learning and Cybernetics*, vol. 3, Qingdao, China, July 2010, pp. 1299–1304.

[18] Y. Noh, Y.-H. Oh, and S.-B. Park, "A location-based personalized news recommendation," in *2014 IEEE International Conference on Big Data and Smart Computing (BIGCOMP)*, Bangkok, Thailand, January 2014, pp. 99–104.

[19] D. S. Chaplot and R. Salakhutdinov, "Knowledge-based word sense disambiguation using topic models," in *32nd AAAI Conference on Artificial Intelligence*, ser. AAAI '18, vol. 32, no. 1, New Orleans, Louisiana USA, February 2018, pp. 5062–5069.

[20] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *Proceedings of the 20th conference on Uncertainty in artificial intelligence*. Banff, Canada: AUAI Press, July 2004, pp. 487–494.

[21] A. McCallum, X. Wang, and A. Corrada-Emmanuel, "Topic and role discovery in social networks with experiments on enron and academic email," *Journal of Artificial Intelligence Research*, vol. 30, no. 7, pp. 249–272, October 2007.

[22] P. N. Howard, G. Bolsover, B. Kollanyi, S. Bradshaw, and L.-M. Neudert, "Junk news and bots during the us election: What were michigan voters sharing over twitter," *CompProp, OII, Data Memo 2017.1*, March 2017. [Online]. Available: https://www.oii.ox.ac.uk/news-events/news/

[23] A. Sharma and U. Ghose, "Sentimental analysis of twitter data with respect to general elections in india," *Procedia Computer Science*, vol. 173, pp. 325–334, June 2020.

[24] S. R. Rufai and C. Bunce, "World leaders' usage of twitter in response to the covid-19 pandemic: a content analysis," *Journal of Public Health*, vol. 42, no. 3, pp. 510–516, April 2020.

[25] X. Wang, L. White, X. Chen, D. D. Gaikar, B. Marakarkandy, and C. Dasgupta, "Using twitter data to predict the performance of bollywood movies," *Industrial Management & Data Systems*, vol. 115, no. 9, pp. 1604–1621, October 2015.

[26] S. Tuarob and C. S. Tucker, "Fad or here to stay: Predicting product market adoption and longevity using large scale, social media data," in *Proceedings of the ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, ser.

IDETC-CIE 2013, vol. Volume 2B: 33rd Computers and Information in Engineering Conference. Portland, Oregon, USA: ASME, August 2013, p. V02BT02A012.

[27] R. L. Gruner, A. Vomberg, C. Homburg, and B. A. Lukas, "Supporting new product launches with social media communication and online advertising: sales volume and profit implications," *Journal of Product Innovation Management*, vol. 36, no. 2, pp. 172–195, 2019.

[28] Z. Zhou, R. Bandari, J. Kong, H. Qian, and V. Roychowdhury, "Information resonance on twitter: Watching iran," in *Proceedings of the First Workshop on Social Media Analytics*, ser. SOMA '10. New York, NY, USA: Association for Computing Machinery, 2010, pp. 123—-131.

[29] P. Garg, H. Garg, and V. Ranga, "Sentiment analysis of the uri terror attack using twitter," in *2017 IEEE International conference on computing, communication and automation (ICCCA)*, Greater Noida, India, May 2017, pp. 17–20.

[30] F. Alam, F. Ofli, and M. Imran, "Crisismmd: Multimodal twitter datasets from natural disasters," in *Proceedings of the International AAAI Conference on Web and Social Media*, ser. AAAI '18, vol. 12, no. 1, New Orleans, Louisiana, USA, February 2018, pp. 465–473.

[31] L. Hong and B. D. Davison, "Empirical study of topic modeling in twitter," in *Proceedings of the first workshop on social media analytics*, Washington DC, Coulumbia, July 2010, pp. 80–88.

[32] W. X. Zhao, J. Jiang, J. Weng, J. He, E.-P. Lim, H. Yan, and X. Li, "Comparing twitter and traditional media using topic models," in *Advances in Information Retrieval*, ser. ECIR 2011, P. Clough, C. Foley, C. Gurrin, G. J. F. Jones, W. Kraaij, H. Lee, and V. Mudoch, Eds. Dublin, Ireland: Springer, April 2011, pp. 338–349.

[33] W. Guo, H. Li, H. Ji, and M. Diab, "Linking tweets to news: A framework to enrich short text data in social media," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, ser. ACL '13, Sofia, Bulgaria, August 2013, pp. 239–249.

[34] P. Li, T. Li, S. Zhang, Y. Li, Y. Tang, and Y. Jiang, "A semi-explicit short text retrieval method combining wikipedia features," *Engineering Applications of Artificial Intelligence*, vol. 94, p. 103809, 2020.

[35] R. Mehrotra, S. Sanner, W. Buntine, and L. Xie, "Improving lda topic models for microblogs via tweet pooling and automatic labeling," in *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '13, Dublin, Ireland, July 2013, pp. 889—-892.

[36] A. Steinskog, J. Therkelsen, and B. Gambäck, "Twitter topic modeling by tweet aggregation," in *Proceedings of the 21st Nordic Conference of Computational Linguistics*, Gothenburg, Sweden, May 2017, pp. 77–86.

[37] Z. Z. Alp and S. G. Öõüdücü, "Extracting topical information of tweets using hashtags," in *14th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Miami, Florida, USA, December 2015, pp. 644–648.

[38] D. Alvarez-Melis and M. Saveski, "Topic modeling in twitter: Aggregating tweets by conversations," in *Proceedings of the 10th International AAAI Conference on Web and Social Media*, Cologne, Germany, May 2016, pp. 519–522.

[39] B. Han, P. Cook, and T. Baldwin, "Lexical normalization for social media text," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 4, no. 1, pp. 1–27, 2013.

[40] B. Tsolmon and K.-S. Lee, "An event extraction model based on timeline and user analysis in latent dirichlet allocation," in *Proceedings of the 37th International ACM SIGIR Conference on Research & Development in Information Retrieval*, ser. SIGIR '14, Gold Coast, Queensland, Australia, July 2014, pp. 1187–1190.

[41] X. Cheng, X. Yan, Y. Lan, and J. Guo, "Btm: Topic modeling over short texts," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 12, pp. 2928–2941, December 2014.

[42] Z. Cheng, J. Caverlee, and K. Lee, "You are where you tweet: A content-based approach to geo-locating twitter users," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, ser. CIKM '10. Toronto, ON, Canada: Association for Computing Machinery, 2010, pp. 759—768.

[43] Y. Wang, J. Liu, J. Qu, Y. Huang, J. Chen, and X. Feng, "Hashtag graph based topic model for tweet mining," in *IEEE International Conference on Data Mining*, Shenzhen, China, December 2014, pp. 1025–1030.

[44] M. H. Alam, W.-J. Ryu, and S. Lee, "Hashtag-based topic evolution in social media," *World Wide Web*, vol. 20, no. 6, pp. 1527–1549, 2017.

[45] Z. Xiaomei, Y. Jing, and Z. Jianpei, "Sentiment-based and hashtag-based chinese online bursty event detection," *Multimedia Tools and Applications*, vol. 77, no. 16, pp. 21 725–21 750, 2018.

[46] Y. Gao, Y. Zhong, D. Preoţiuc-Pietro, and J. J. Li, "Predicting and analyzing language specificity in social media posts," in *Proceedings of the AAAI Conference on Artificial Intelligence*, ser. AAAI-19, vol. 33, no. 01, Honolulu, Hawaii, USA, January 2019, pp. 6415–6422.

[47] L. AlSumait, D. Barbará, and C. Domeniconi, "On-line lda: Adaptive topic models for mining text streams with applications to topic detection and tracking," in *8th IEEE International Conference of Data Mining*, ser. ICDM '08, Pisa,Italy, December 2008, pp. 3–12.

[48] C. C. Aggarwal and H. Wang, "Text mining in social networks," in *Social Network Data Analytics*, 1st ed., C. C. Aggarwal, Ed. Boston, USA: Springer, 2011, pp. 353–378.

[49] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman, "Indexing by latent semantic analysis," *Journal of the American society for Information Science*, vol. 41, no. 6, pp. 391–407, September 1990.

[50] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, October 1999.

[51] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine learning*, vol. 42, no. 1, pp. 177–196, January 2001.

[52] A. E. Gelfand, "Gibbs sampling," *Journal of the American statistical Association*, vol. 95, no. 452, pp. 1300–1304, February 2000.

[53] G. Heinrich, "Parameter estimation for text analysis," Fraunhofer Institute for Computer Graphics Research IGD, Tech. Rep. 2.9, September 2009. [Online]. Available: https://www.arbylon.net/publications/text-est2.pdf

[54] H. Schütze, C. D. Manning, and P. Raghavan, *Introduction to information retrieval*. Cambridge University Press Cambridge, 2008, vol. 39.

[55] J. Chang, J. Boyd-Graber, S. Gerrish, C. Wang, and D. M. Blei, "Reading tea leaves: How humans interpret topic models," in *Proceedings of the 22nd*

International Conference on Neural Information Processing Systems, ser. NIPS'09, Red Hook, NY, USA, December 2009, p. 288–296.

[56] A. B. Dieng, F. J. Ruiz, and D. M. Blei, "Topic modeling in embedding spaces," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 439–453, July 2020.

[57] Y. Tu, N. Johri, D. Roth, and J. Hockenmaier, "Citation author topic model in expert search," in *Proceedings of the 23rd International Conference on Computational Linguistics*, ser. COLING 2010, Beijing, China, August 2010, pp. 1265–1273.

[58] X. Wang and A. McCallum, "Topics over time: a non-markov continuous-time model of topical trends," in *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '06, Philadelphia PA, USA, August 2006, pp. 424–433.

[59] D. M. Blei and J. D. Lafferty, "Dynamic topic models," in *Proceedings of the 23rd International Conference on Machine Learning*, ser. ICML '06. Pittsburgh, Pennsylvania, USA: ACM, June 2006, pp. 113–120.

[60] C. Wang, D. Blei, and D. Heckerman, "Continuous time dynamic topic models," *arXiv:1206.3298*, June 2012.

[61] E. Cinlar, *Introduction to stochastic processes*. Mineola, New York: DOVER PUBLICATIONS INC., 2013.

[62] C.-C. Pan and P. Mitra, "Event detection with spatial latent dirichlet allocation," in *Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries*, ser. JCDL '11, Ottawa, Ontario, Canada, June 2011, pp. 349–358.

[63] D. Graff, C. Cieri, S. Strassel, and N. Martey, "The tdt-3 text and speech corpus," in *Proceedings of DARPA Broadcast News Workshop*, 1999, pp. 57–60. [Online]. Available: https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/tdt2000-tdt3-corpus.pdf

[64] D. Ramage, D. Hall, R. Nallapati, and C. D. Manning, "Labeled lda: A supervised topic model for credit attribution in multi-labeled corpora," in *Proceedings of the 2009 Conference on EMNLP: Volume 1*, ser. EMNLP '09, Singapore, August 2009, pp. 248–256.

[65] D. M. Blei and J. D. McAuliffe, "Supervised topic models," in *Proceedings of the 20th International Conference on Neural Information Processing Systems*, ser. NIPS '07, Vancouver, British Columbia, Canada, December 2007, pp. 121–128.

[66] S. Lacoste-Julien, F. Sha, and M. I. Jordan, "Disclda: Discriminative learning for dimensionality reduction and classification," in *Proceedings of the 21st International Conference on Neural Information Processing Systems*, ser. NIPS '08, Vancouver, British Columbia, Canada, December 2008, pp. 897—904.

[67] J. Zhu, A. Ahmed, and E. P. Xing, "Medlda: Maximum margin supervised topic models," *Journal of Machine Learning Research*, vol. 13, no. 74, pp. 2237–2278, August 2012.

[68] D. Kim, S. Kim, and A. Oh, "Dirichlet process with mixed random measures: A nonparametric topic model for labeled data," in *Proceedings of the 29th International Conference on Machine Learning*, ser. ICML '12, J. Langford and J. Pineau, Eds., Edinburgh, Scotland, UK, July 2012, pp. 727–734.

[69] T. N. Rubin, A. Chambers, P. Smyth, and M. Steyvers, "Statistical topic models for multi-label document classification," *Machine learning*, vol. 88, no. 1-2, pp. 157–208, December 2012.

[70] M. Jankowski, "Boost multi-class slda model for text classification," in *Artificial Intelligence and Soft Computing*, L. Rutkowski, R. Scherer, M. Korytkowski, W. Pedrycz, R. Tadeusiewicz, and J. M. Zurada, Eds. Cham: Springer International Publishing, May 2018, pp. 633–644.

[71] J. Wood, P. Tan, W. Wang, and C. Arnold, "Source-lda: Enhancing probabilistic topic models using prior knowledge sources," in *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, San Diego, California, USA, April 2017, pp. 411–422.

[72] J. Jagarlamudi, H. Daumé III, and R. Udupa, "Incorporating lexical priors into topic models," in *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, ser. EACL '12, Avignon, France, April 2012, pp. 204–213.

[73] Q. Diao, J. Jiang, F. Zhu, and E. P. LIM, "Finding bursty topics from microblogs," in *Proceedings of the 50th Annual Meeting of the Association*

for Computational Linguistics, ser. ACL '12, Jeju Island, Korea, July 2012, pp. 536–544.

[74] X. Yan, J. Guo, Y. Lan, and X. Cheng, "A biterm topic model for short texts," in *Proceedings of the 22nd International Conference on World Wide Web*, ser. WWW '13, Rio de Janeiro, Brazil, May 2013, pp. 1445—1456.

[75] Y. Wang, J. Liu, Y. Huang, and X. Feng, "Using hashtag graph-based topic model to connect semantically-related words without co-occurrence in microblogs," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 7, pp. 1919–1933, 2016.

[76] C. Xing, Y. Wang, J. Liu, Y. Huang, and W.-Y. Ma, "Hashtag-based sub-event discovery using mutually generative lda in twitter," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, ser. AAAI '16, Phoenix, Arizona, February 2016, pp. 2666–2672.

[77] D. Davidov, O. Tsur, and A. Rappoport, "Enhanced sentiment learning using twitter hashtags and smileys," in *Proceedings of the 23rd International Conference on Computational Linguistics*, ser. COLING '10, Beijing, China, August 2010, pp. 241–249.

[78] S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen, "Microblogging during two natural hazards events: What twitter may contribute to situational awareness," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '10, Atlanta, Georgia, USA, Aril 2010, pp. 1079–1088.

[79] A. Cui, M. Zhang, Y. Liu, S. Ma, and K. Zhang, "Discover breaking events with popular hashtags in twitter," in *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, ser. CIKM '12, Maui, Hawaii, USA, November 2012, pp. 1794–1798.

[80] X. Chen, X. Zhou, J. Chan, L. Chen, T. Sellis, and Y. Zhang, "Event popularity prediction using influential hashtags from social media," *IEEE Transactions on Knowledge and Data Engineering*, December 2020.

[81] M. S. C. Sapul, T. H. Aung, and R. Jiamthapthaksin, "Trending topic discovery of twitter tweets using clustering and topic modeling algorithms," in *2017 14th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, Nakhon Si Thammarat, Thailand, July 2017, pp. 1–6.

[82] I. Alfina, D. Sigmawaty, F. Nurhidayati, and A. N. Hidayanto, "Utilizing hashtags for sentiment analysis of tweets in the political domain," in *Proceedings of the 9th International Conference on Machine Learning and Computing*, ser. ICMLC 2017, Singapore, February 2017, pp. 43–47.

[83] Y. Gao, J. Sang, T. Ren, and C. Xu, "Hashtag-centric immersive search on social media," in *Proceedings of the 25th ACM International Conference on Multimedia*, ser. MM '17, Mountain View, California, USA, October 2017, pp. 1924–1932.

[84] S. Shen, N. Murzintcev, C. Song, and C. Cheng, "Information retrieval of a disaster event from cross-platform social media," *Information Discovery and Delivery*, vol. 45, no. 4, pp. 220–226, November 2017.

[85] F. Abel, Q. Gao, G.-J. Houben, and K. Tao, "Analyzing user modeling on twitter for personalized news recommendations," in *Proceedings of the 19th International Conference on User Modeling, Adaption, and Personalization*, ser. UMAP'11, Girona, Spain, July 2011, pp. 1—-12.

[86] P. M. A. Kumar, K. Charan, G. V. S. Kumar, K. Amith, and K. Krishna, "Real-time hashtag based event detection model with sentiment analysis for recommending user tweets," in *3rd IEEE International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV)*, Tirunelveli, India, February 2021, pp. 1437–1444.

[87] D. M. Romero, B. Meeder, and J. Kleinberg, "Differences in the mechanics of information diffusion across topics: Idioms, political hashtags, and complex contagion on twitter," in *Proceedings of the 20th International Conference on World Wide Web*, ser. WWW '11, Hyderabad, India, March 2011, pp. 695—-704.

[88] O. Oh, M. Agrawal, and H. R. Rao, "Information control and terrorism: Tracking the mumbai terrorist attack through twitter," *Information Systems Frontiers*, vol. 13, no. 1, pp. 33–43, September 2011.

[89] D. Mair, "# westgate: A case study: How al-shabaab used twitter during an ongoing attack," *Studies in conflict & terrorism*, vol. 40, no. 1, pp. 24–43, February 2017.

[90] B. Truong, C. Caragea, A. Squicciarini, and A. H. Tapia, "Identifying valuable information from twitter during natural disasters," *Proceedings of the*

American Society for Information Science and Technology, vol. 51, no. 1, pp. 1–4, April 2014.

[91] N. Pourebrahim, S. Sultana, J. Edwards, A. Gochanour, and S. Mohanty, "Understanding communication dynamics on twitter during natural disasters: A case study of hurricane sandy," *International Journal of Disaster Risk Reduction*, vol. 37, p. 101176, July 2019.

[92] F. Toriumi, T. Sakaki, K. Shinoda, K. Kazama, S. Kurihara, and I. Noda, "Information sharing on twitter during the 2011 catastrophic earthquake," in *Proceedings of the 22nd International Conference on World Wide Web*, ser. WWW '13, Rio de Janeiro, Brazil, May 2013, pp. 1025—1028.

[93] T. Hennig-Thurau, C. Wiertz, and F. Feldhaus, "Does twitter matter? the impact of microblogging word of mouth on consumers' adoption of new movies," *Journal of the Academy of Marketing Science*, vol. 43, no. 3, pp. 375–394, June 2015.

[94] S. Ahmed, K. Jaidka, and J. Cho, "The 2014 indian elections on twitter: A comparison of campaign strategies of political parties," *Telematics and Informatics*, vol. 33, no. 4, pp. 1071–1087, 2016.

[95] P. Sharma and T.-S. Moh, "Prediction of indian election using sentiment analysis on hindi twitter," in *2016 IEEE International Conference on Big Data (Big Data)*, Washington, DC, USA, December 2016, pp. 1966–1971.

[96] A. Signorini, A. M. Segre, and P. M. Polgreen, "The use of twitter to track levels of disease activity and public concern in the u.s. during the influenza a h1n1 pandemic," *PLOS ONE*, vol. 6, no. 5, pp. 1–10, May 2011.

[97] V. K. Jain and S. Kumar, "An effective approach to track levels of influenza-a (h1n1) pandemic in india using twitter," *Procedia Computer Science*, vol. 70, pp. 801–807, 2015.

[98] D. Wang, S. Zhu, T. Li, and Y. Gong, "Multi-document summarization using sentence-based topic models," in *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, ser. ACLShort '09, Suntec, Singapore, August 2009, pp. 297—-300.

[99] G. Yang, D. Wen, N.-S. Chen, E. Sutinen *et al.*, "A novel contextual topic model for multi-document summarization," *Expert Systems with Applications*, vol. 42, no. 3, pp. 1340–1352, February 2015.

[100] E. Y. Hidayat, F. Firdausillah, K. Hastuti, I. N. Dewi, and A. Azhari, "Automatic text summarization using latent drichlet allocation (lda) for document clustering," *International Journal of Advances in Intelligent Informatics*, vol. 1, no. 3, pp. 132–139, 2015.

[101] X. Chen, Y. Xia, P. Jin, and J. Carroll, "Dataless text classification with descriptive lda," in *Proceedings of the AAAI Conference on Artificial Intelligence*, ser. AAAI'15, Austin, Texas, January 2015, pp. 2224—-2231.

[102] L. Liefa and Z. Y. Le Fugang, "The application of lda model in patent text classification," *Journal of Modern Information*, vol. 37, no. 3, pp. 35–39, 2017.

[103] A. Chaney and D. Blei, "Visualizing topic models," in *Proceedings of the International AAAI Conference on Web and Social Media*, ser. ICWSM '12, vol. 6, no. 1, Dublin, Ireland, June 2012, pp. 419–422.

[104] N. Thapen, D. Simmie, and C. Hankin, "The early bird catches the term: combining twitter and news data for event detection and situational awareness," *Journal of biomedical semantics*, vol. 7, no. 1, p. 61, October 2016.

[105] S. Zhang, D. Zheng, X. Hu, and M. Yang, "Bidirectional long short-term memory networks for relation classification," in *Proceedings of the 29th Pacific Asia Conference on Language, Information and Computation*, Shanghai, China, October 2015, pp. 73–78.

[106] A. Graves, "Supervised sequence labelling with recurrent neural networks," Ph.D. dissertation, Technical University of Munich, Munich, Germany, 2008.

[107] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, ser. NAACL-HLT 2019, Minneapolis, Minnesota, June 2019, pp. 4171–4186.

[108] M. Zhang, Z. Cui, M. Neumann, and Y. Chen, "An end-to-end deep learning architecture for graph classification," in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*, ser. AAAI '18, vol. 32, no. 1, New Orleans, Louisiana, USA., February 2018, pp. 4438–4445.

[109] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *5th International Conference on Learning Representations*, ser. ICLR 2017, Toulon, France, April 2017, pp. 1–14.

[110] C. Lu, W. Lam, and Y. Zhang, "Twitter user modeling and tweets recommendation based on wikipedia concept graph," in *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, ser. AAAI '12, Toronto, Ontario, Canada, July 2012, pp. 33–38.

[111] J. Kang and H. Lee, "Modeling user interest in social media using news media and wikipedia," *Information Systems*, vol. 65, pp. 52–64, April 2017.

[112] S. A. P. Romero and K. Becker, "Experiments with semantic enrichment for event classification in tweets," in *2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, Omaha, NE, USA, October 2016, pp. 503–506.

[113] K. Morabia, N. L. B. Murthy, A. Malapati, and S. Samant, "Sedtwik: segmentation-based event detection from tweets using wikipedia," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, ser. NAACL 2019, Minneapolis, Minnesota, June 2019, pp. 77–85.

[114] A. Gattani, D. S. Lamba, N. Garera, M. Tiwari, X. Chai, S. Das, S. Subramaniam, A. Rajaraman, V. Harinarayan, and A. Doan, "Entity extraction, linking, classification, and tagging for social media: A wikipedia-based approach," *Proceedings of the VLDB Endowment*, vol. 6, no. 11, pp. 1126––1137, August 2013.

[115] C. Li, A. Sun, J. Weng, and Q. He, "Tweet segmentation and its application to named entity recognition," *IEEE Transactions on Knowledge and Data Engineering*, vol. 27, no. 2, pp. 558–570, 2015.

[116] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis," in *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, ser. IJCAI'07, vol. 7, January 2007, pp. 1606–1611.

[117] R. Jarvis and E. Patrick, "Clustering using a similarity measure based on shared near neighbors," *IEEE Transactions on Computers*, vol. C-22, no. 11, pp. 1025–1034, 1973.

[118] G. Erkan and D. R. Radev, "Lexrank: Graph-based lexical centrality as salience in text summarization," *Journal of Artificial Intelligence Research*, vol. 22, no. 1, pp. 457—479, December 2004.

[119] A. Ollagnier and H. Williams, "Network-based pooling for topic modeling on microblog content," in *International Symposium on String Processing and Information Retrieval*, Exter, UK, October 2019, pp. 80–87.

[120] M. Hajjem and C. Latiri, "Combining ir and lda topic modeling for filtering microblogs," *Procedia Computer Science*, vol. 112, pp. 761–770, September 2017.

[121] E. Zangerle, W. Gassler, and G. Specht, "Recommending #-tags in twitter," in *2nd International Workshop on Semantic Adaptive Social Web*, ser. SASWeb 2011, Girona, Spain, July 2011, pp. 67–78.

[122] T. Li, Y. Wu, and Y. Zhang, "Twitter hash tag prediction algorithm," in *Proceedings on the International Conference on Internet Computing (ICOMP)*, Las Vegas, Nevada, USA, July 2011, p. 1.

[123] R. Krestel, P. Fankhauser, and W. Nejdl, "Latent dirichlet allocation for tag recommendation," in *Proceedings of the Third ACM Conference on Recommender Systems*, ser. RecSys '09, New York, NY, USA, October 2009, p. 61–68.

[124] Z. Liu, X. Chen, and M. Sun, "A simple word trigger method for social tag suggestion," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, ser. EMNLP '11, Edinburgh, Scotland, United Kingdom, July 2011, pp. 1577—1588.

[125] Z. Ding, Q. Zhang, and X.-J. Huang, "Automatic hashtag recommendation for microblogs using topic-specific translation model," in *Proceedings of COLING 2012*. Citeseer, December 2012, pp. 265–274.

[126] Z. Ding, X. Qiu, Q. Zhang, and X. Huang, "Learning topical translation model for microblog hashtag suggestion," in *Proceedings of the 23rd International Joint Conference on Artificial Intelligence*, ser. IJCAI '13, Beijing, China, August 2013, pp. 2078—2084.

[127] A. Tomar, F. Godin, B. Vandersmissen, W. De Neve, and R. Van de Walle, "Towards twitter hashtag recommendation using distributed word representations and a deep feed forward neural network," in *IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Delhi,India, 2014, pp. 362–368.

[128] J. Li, H. Xu, X. He, J. Deng, and X. Sun, "Tweet modeling with lstm recurrent neural networks for hashtag recommendation," in *2016 International Joint Conference on Neural Networks (IJCNN)*, Vancouver, BC, Canada, July 2016, pp. 1570–1577.

[129] S. M. Kywe, T.-A. Hoang, E.-P. Lim, and F. Zhu, "On recommending hashtags in twitter networks," in *International Conference on Social Informatics*, K. Aberer, A. Flache, W. Jager, L. Liu, J. Tang, and C. Guéret, Eds., Lausanne, Switzerland, December 2012, pp. 337–350.

[130] Y. Gong and Q. Zhang, "Hashtag recommendation using attention-based convolutional neural network," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, ser. IJCAI'16, New York, New York, USA, July 2016, pp. 2782—2788.

[131] A. Leman and B. Weisfeiler, "A reduction of a graph to a canonical form and an algebra arising during this reduction," *Nauchno-Technicheskaya Informatsiya*, vol. 2, no. 9, pp. 12–16, 1968.

[132] S. Temma, M. Sugii, and H. Matsuno, "The document similarity index based on the jaccard distance for mail filtering," in *34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, JeJu, South Korea, 2019, pp. 1–4.

[133] J. Qiang, P. Chen, T. Wang, and X. Wu, "Topic modeling over short texts by incorporating word embeddings," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, ser. PAKDD 2017, J. Kim, K. Shim, L. Cao, J.-G. Lee, X. Lin, and Y.-S. Moon, Eds., Jeju, South Korea, May 2017, pp. 363–374.

[134] N. Peinelt, D. Nguyen, and M. Liakata, "tbert: Topic models and bert joining forces for semantic similarity detection," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, July 2020, pp. 7047–7055.

[135] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, and J. Gao, "Deep learning–based text classification: A comprehensive review," *ACM Computing Surveys (CSUR)*, vol. 54, no. 3, pp. 1–40, April 2021.

[136] F. Hamborg, C. Breitinger, M. Schubotz, S. Lachnit, and B. Gipp, "Extraction of main event descriptors from news articles by answering the journalistic five w and one h questions," in *Proceedings of the 18th ACM/IEEE*

on Joint Conference on Digital Libraries, ser. JCDL '18, Fort Worth Texas USA, May 2018, pp. 339–340.

[137] J. Zhang and Y. Luo, "Degree centrality, betweenness centrality, and closeness centrality in social network," in *2017 2nd International Conference on Modelling, Simulation and Applied Mathematics (MSAM2017)*, ser. Advances in Intelligent System Research, March 2017, pp. 300–303.

[138] J. R. Finkel, T. Grenager, and C. Manning, "Incorporating non-local information into information extraction systems by gibbs sampling," in *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, ser. ACL'05, Ann Arbor, Michigan, 2005, pp. 363–370.

[139] A. Clauset, C. Moore, and M. E. Newman, "Hierarchical structure and the prediction of missing links in networks," *Nature*, vol. 453, no. 7191, pp. 98–101, May 2008.

[140] M. K. Sparrow, "The application of network analysis to criminal intelligence: An assessment of the prospects," *Social Networks*, vol. 13, no. 3, pp. 251–274, September 1991.

[141] V. Krebs, "Mapping networks of terrorist cells," *CONNECTIONS*, vol. 24, no. 3, pp. 43–52, 2002.

[142] R. M. Medina, "Social network analysis: a case study of the islamist terrorist network," *Security Journal*, vol. 27, no. 1, pp. 97–121, February 2014.

[143] K. M. Carley, J. Reminga, and N. Kamneva, "Destabilizing terrorist networks," in *Proceedings of North American Association for Computational, Social, and Organizational Sciences*, ser. NAACSOS '03, Pittsburgh, PA, 2003.

[144] F. Spezzano, V. S. Subrahmanian, and A. Mannes, "Reshaping terrorist networks," *Communications of the ACM*, vol. 57, no. 8, pp. 60—69, August 2014.

[145] "National consortium for the study of terrorism and responses to terrorism (start). global terrorism database [data file]."

[146] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (roc) curve," *Radiology*, vol. 143, no. 1, pp. 29–36, April 1982.

[147] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proceedings of the 26th International Conference on Neural Information Processing Systems*, ser. NIPS '13, Lake Tahoe, Nevada, December 2013, pp. 3111—-3119.

[148] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," *arXiv:1712.09405*, December 2017.

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

Publications (Related to Thesis)

Conference:

1. Anil, A., **Kumar, D.**, Sharma S., Singha, R., Sarmah, R., Ranjan, Bhattacharya, N. and Singh, S.R. **Link prediction using social network analysis over heterogeneous terrorist network**. In *2015 IEEE International Conference on Smart City/ SocialCom/SustainCom (SmartCity)*, 2015, pages 267–272.

2. **Kumar, D.** and Singh, S.R. **Prioritized Named Entity Driven LDA for Document Clustering**. In *International Conference on Pattern Recognition and Machine Intelligence*, Springer, 2019, pages 294–301.

3. **Kumar, D.** and Singh, S.R. **Prioritizing Hashtags for Improved Topic Modeling over Tweets**. [Under Review]. [1]

Journal:

1. **Kumar, D.**, Singh, L.G. and Singh., S.R. **Hashtag based semantic expansion of Tweets for improved Topic modeling**. In *IEEE Transaction on Computational Social System* [2^{nd} Major revision submitted on 26^{th} February 2022].

Other Publications

Conference:

1. Kumar, S., **Kumar, D.**, Singh., S.R. **Detecting Fake News Articles Through Hierarchical Encoding**. [Under Review].

2. Mitra, A., Singh, L.G., Singh, R., **Kumar, D.**, Singh., S.R., and Kumar S. **Correlating Social Network Activity and Physical Network Activity through Mobility Patterns and Sentiments**. [Under Review].

[1] The venue of some of the publications has not been given due to double-blind review policy.

Journal:

1. Naskar, D. Singh, S.R., **Kumar, D.**, Nandi, S., and Rivaherrera, E.O. . **Emotion Dynamics of Public Opinions on Twitter**. In *ACM Transactions on Information Systems (TOIS)*, vol. 38, no. 2, pp. 1–24, 2020.

2. Singh, A. K., Singh, S.R., **Kumar, D.**. **Understanding the effect of Embedding on Missing Trajectory Prediction using Sequence-to-Sequence models and Geohash encoded vessel trajectories.** In *The Journal of Navigation* [Under Review].

Brief Biography of the Author

Durgesh Kumar was born in Godda, Jharkhand, India on 16th December 1989. After completing his basic education from Banka, Bihar, he completed Bachelor of Technology (B.Tech) in Dept. of Information Technology from Haldia Institute of Technology, Haldia, West Bengal in the year 2012. He was enrolled as a Ph.D. research scholar in the Dept. of Computer Science & Engineering at Indian Institute of Technology, Guwahati. In Ph.D., he was supervised by Dr. Sanasam Ranbir Singh. His research interests include Topic Modeling, Natural Language Processing, Social Network Analysis, Machine Learning, and Deep Learning.