Copyright
by
Nazneen Fatema Naushad Rajani
2014
New Topic Detection in Microblogs and Topic Model Evaluation using Topical Alignment

APPROVED BY

SUPERVISING COMMITTEE:

__________________________________________
Jason Baldridge, Supervisor

__________________________________________
Pradeep Ravikumar
New Topic Detection in Microblogs and Topic Model Evaluation using Topical Alignment

by

Nazneen Fatema Naushad Rajani, MSc.Tech

THESIS
Presented to the Faculty of the Graduate School of
The University of Texas at Austin
in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science in Computer Science

THE UNIVERSITY OF TEXAS AT AUSTIN
May 2014
Dedicated to my loving mother Gulshan Rajani and father Naushad Rajani.
Acknowledgments

Firstly I am thankful to God for everything that has enabled me to do quality research and gain immense knowledge through learning. Secondly to my parents I owe what I am today and it is only because of their support and encouragement that this has been possible.

I am grateful to my Supervisor, Prof. Jason Baldridge for being open to my ideas and guiding me throughout my research journey. I am thankful to Prof. Pradeep Ravikumar for his valuable feedback on my thesis.

My brother, sister and my loving husband for being a source of my happiness.
This thesis deals with topic model evaluation and new topic detection in microblogs. Microblogs are short and thus may not carry any contextual clues. Hence it becomes challenging to apply traditional natural language processing algorithms on such data. Graphical models have been traditionally used for topic discovery and text clustering on sets of text-based documents. Their unsupervised nature allows topic models to be trained easily on datasets meant for specific domains. However the advantage of not requiring annotated data comes with a drawback with respect to evaluation difficulties. The problem aggravates when the data comprises microblogs which are unstructured and noisy.

We demonstrate the application of three types of such models to microblogs - the Latent Dirichlet Allocation, the Author-Topic and the Author-Recipient-Topic model. We extensively evaluate these models under different
settings, and our results show that the Author-Recipient-Topic model extracts the most coherent topics. We also addressed the problem of topic modeling on short text by using clustering techniques. This technique helps in boosting the performance of our models.

Topical alignment is used for large scale assessment of topical relevance by comparing topics to manually generated domain specific concepts. In this thesis we use this idea to evaluate topic models by measuring misalignments between topics. Our study on comparing topic models reveals interesting traits about Twitter messages, users and their interactions and establishes that joint modeling on author-recipient pairs and on the content of tweet leads to qualitatively better topic discovery.

This thesis gives a new direction to the well known problem of topic discovery in microblogs. Trend prediction or topic discovery for microblogs is an extensive research area. We propose the idea of using topical alignment to detect new topics by comparing topics from the current week to those of the previous week. We measure correspondence between a set of topics from the current week and a set of topics from the previous week to quantify five types of misalignments: junk, fused, missing and repeated. Our analysis compares three types of topic models under different settings and demonstrates how our framework can detect new topics from topical misalignments. In particular so-called junk topics are more likely to be new topics and the missing topics are likely to have died or die out.

To get more insights into the nature of microblogs we apply topical
alignment to hashtags. Comparing topics to hashtags enables us to make interesting inferences about Twitter messages and their content. Our study revealed that although a very small proportion of Twitter messages explicitly contain hashtags, the proportion of tweets that discuss topics related to hashtags is much higher.
# Table of Contents

Acknowledgments .......................................................... v

Abstract ................................................................. vi

List of Tables ........................................................... xi

List of Figures .......................................................... xii

Chapter 1. Introduction ..................................................... 1
  1.1 Microblogs .......................................................... 2
  1.2 Research Motivation .................................................. 2
  1.3 Thesis Contribution ................................................... 4
  1.4 Publications .......................................................... 6

Chapter 2. Background and Related Work ............................. 7
  2.1 Background ........................................................... 7
    2.1.1 Topic Models and their Characteristics ......................... 9
  2.2 Related Work ......................................................... 10
    2.2.1 Topic Models for Information Discovery ....................... 10
    2.2.2 Topic Models for Text Categorization ......................... 11
    2.2.3 Evaluating Topic Models ...................................... 12
    2.2.4 Topic Models and Online Social Media ....................... 12

Chapter 3. System Design and Implementation ....................... 13
  3.1 System Design ....................................................... 13
  3.2 Dataset ............................................................ 18
    3.2.1 Filtering on Tweets ............................................. 18
    3.2.2 Filtering on Word Tokens ..................................... 22
  3.3 Implementation ..................................................... 23
<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>List of stop words.</td>
<td>23</td>
</tr>
<tr>
<td>3.2</td>
<td>Probabilities that a unigram is associated with Topic 1 or Topic 2.</td>
<td>25</td>
</tr>
<tr>
<td>3.3</td>
<td>In a correspondence matrix, each entry $p_{r,l}$ represents the probability that a reference topic $r$ and a latent topic $l$ are equivalent. Misalignment scores measure how much topical alignment deviates from an optimal one-to-one correspondence. Comparing a latent topic to all reference topics, <em>junk</em> and <em>fused</em> scores measure how likely the topic matches exactly zero, or more than one reference topic. <em>Missing</em> and <em>repeated</em> scores measure how likely a reference topic matches exactly zero, or more than one latent topic.</td>
<td>26</td>
</tr>
<tr>
<td>4.1</td>
<td>Top 10 words belonging to topic related to “Austin” for each of the LDA, AT and ART topic models.</td>
<td>33</td>
</tr>
<tr>
<td>4.2</td>
<td>PMI scores for LDA, AT and ART models trained on the Single-Tweet and Clustered Datasets, with 500 iterations.</td>
<td>33</td>
</tr>
<tr>
<td>4.3</td>
<td>PMI scores for LDA, AT and ART models trained on the Single-Tweet and Clustered Datasets, with 1000 iterations.</td>
<td>34</td>
</tr>
<tr>
<td>4.4</td>
<td>Top 10 words from topics that were missing for week 2 and junk for week 3 for AT and ART models respectively.</td>
<td>43</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>System design overview.</td>
<td>14</td>
</tr>
<tr>
<td>3.2</td>
<td>LDA, AT, and ART models. Modified from [19].</td>
<td>15</td>
</tr>
<tr>
<td>3.3</td>
<td>Example of association of words to topics.</td>
<td>18</td>
</tr>
<tr>
<td>3.4</td>
<td>Example of a conversation on Twitter.</td>
<td>20</td>
</tr>
<tr>
<td>3.5</td>
<td>Tweets containing #sxsw pooled together to form a Clustered Dataset.</td>
<td>22</td>
</tr>
<tr>
<td>3.6</td>
<td>Step-by-step overview of new topic prediction for microblogs.</td>
<td>28</td>
</tr>
<tr>
<td>4.1</td>
<td>Normalized PMI scores for LDA, AT and ART on the Single-Tweet Dataset (500 iterations).</td>
<td>35</td>
</tr>
<tr>
<td>4.2</td>
<td>Normalized PMI scores for LDA, AT, and ART on the Clustered Dataset (500 iterations).</td>
<td>35</td>
</tr>
<tr>
<td>4.3</td>
<td>Correspondence matrix between latent AT or ART topic vectors and reference LDA topic vectors. Color intensity shows the likelihood that reference topics and latent topics are equivalent. The x-axis shows the number of LDA topics and y-axis shows the number of AT topics and ART topics respectively.</td>
<td>37</td>
</tr>
<tr>
<td>4.4</td>
<td>Topical alignment to LDA reference topics for ( N \in [100, 600] ) topics(x-axis) and ( \alpha = 5/N ) and ( \beta = 0.25 ). The y-axis shows the fraction of AT or ART topics that have a single matching(resolved), multiple matching LDA topics(repeated) or are subsumed by one(fused) or multiple fused LDA topics(fused and repeated).</td>
<td>38</td>
</tr>
<tr>
<td>4.5</td>
<td>Percentage of AT and ART resolved topics to LDA topics, x-axis represents ( N \in [100, 600] ).</td>
<td>39</td>
</tr>
<tr>
<td>4.6</td>
<td>Topical alignment for weekly AT topics for ( N \in [100, 600] ) topics(x-axis) and ( \alpha = 5/N ) and ( \beta = 0.25 ). The y-axis shows the fraction of a week’s topic that have a single matching(resolved), multiple matching (repeated) to previous week’s topics or are subsumed by one(fused) or multiple fused topics of previous week(fused and repeated).</td>
<td>41</td>
</tr>
</tbody>
</table>
4.7 Topical alignment for weekly ART topics for $N \in [100, 600]$ topics (x-axis) and $\alpha = 5/N$ and $\beta = 0.25$. The y-axis shows the fraction of a week’s topic that have a single matching (resolved), multiple matching (repeated) to previous week’s topics or are subsumed by one (fused) or multiple fused topics of previous week (fused and repeated). ................................................. 42

4.8 Twitter trends for 2009. ................................................. 43

4.9 Topical alignment hashtag vectors for $N \in [100, 600]$ (x-axis) and $\alpha = 5/N$ and $\beta = 0.25$. The y-axis shows the fraction of latent topics that have a single matching (resolved), multiple matching hashtags (repeated) or are subsumed by one (fused) or multiple fused hashtags (fused and repeated). .................. 45

4.10 Topical alignment hashtag vectors for $N \in [100, 600]$ (x-axis) and $\alpha = 5/N$ and $\beta = 0.25$. The y-axis shows the fraction of latent topics that have a single matching (resolved), multiple matching hashtags (repeated) or are subsumed by one (fused) or multiple fused hashtags (fused and repeated). The LDA topical alignment on the right side is the same as in Figure 4.9 and is repeated here for comparison with the ART model. .................. 46

4.11 Percentage of resolved topics for AT and ART models, x-axis represents $N \in [100, 600]$. ................................................. 46
Chapter 1

Introduction

A part of this thesis deals with extracting meaningful topics from microblogs using unsupervised graphical models and evaluating their performance with and without clustering the data. The other part explores detection of new topics in microblogs using the idea of topical alignment.

We propose using the Author-Topic(AT) [26] and the Author-Recipient-Topic(ART) [19] models for microblogs and compare it to the well known Latent Dirichlet Allocation(LDA) based on several parameters. Automatic validation of latent topics is a hard problem and thus we perform topic model diagnostics using the idea of topic alignment between reference and latent topics.

In this chapter we present an introduction to microblogs, topic extraction from data and the need to analyze and evaluate topics from online social microblogs. We highlight the importance of topic model diagnostics while using unsupervised models and present a formal thesis definition.
1.1 Microblogs

Microblogs such as Twitter are a type of online social media systems that allow users to post short messages called status updates to their homepage. Status updates from Twitter are called tweets and they are often related to an event or the user’s specific interest topic or his/her personal thoughts and opinions. A word, phrase or topic that is tagged (contained in a tweet) at a greater rate than other tags is said to be a trending topic. Trending topics become popular either through a concerted effort by users, or because of an event that prompts people to talk about one specific topic 1.

Twitter has gained lot of importance due to its ability to disseminate information rapidly and more so during events related to natural disasters, political turmoil or other such crises. Researchers are actively analyzing such micro-blogging systems and search engines like Google have started including tweets in their search results.

1.2 Research Motivation

With an average of 10000 tweets currently generated per second, analyzing them to understand what topics they discuss is an important research study. However, applying traditional natural language processing techniques that use syntactic and semantic models on such data is challenging mainly due to following reasons [14].

\(^{1}\)http://en.wikipedia.org/wiki/Twitter#cite_note-105
• Tweets are short in length with a 140-character limit and thus may not carry many contextual clues about the content’s subject matter.

• Tweets are very informally written, often very unstructured and consisting of ungrammatical text.

• Tweets may contain implied references to locations or things, thus making named entity recognition difficult [27].

Topic models are generative models and a popular approach for modeling term frequency occurrences in documents in a given corpus. The basic approach of topic modeling is to describe a document as mixture of different topics. Such models can be helpful in developing systems that can help in analyzing the content of a stream of text. Using such a system would enable us to identify the topic or event that a particular tweet is about. Thus using such a system on large datasets would not only alert observers to crisis situations such as diplomatic tensions or upcoming revolution ahead of time but also predict trending topics and thus help in targeting ads and making appropriate recommendations to users.

Since most of the user data available is not labeled, it is hard to evaluate systems that learn from such data to make predictions or recommendations. After the model is created, spot-checking of topics in an ad-hoc manner may be used to analyze topic relevance. To overcome the problems of topic model diagnostics with minimal human intervention, we present the idea of automatically evaluating models by calculating topical alignment between a set of
latent topics to a set of reference topics. Such a system would be scalable to large data and usable for topics in any domain.

Further we introduce the idea of topic discovery in microblogs using topical alignment between topics from the current week to those of previous week. We propose ideas to discover topics and also to detect when topics die out. Finally we evaluate our system on tweets and assess manually to strengthen our findings.

1.3 Thesis Contribution

The thesis contribution can be briefly stated as:

1. We introduce the use of Author-Topic (AT) model and the Author-Recipient-Topic (ART) model approaches for microblogs and compare it to the Latent Dirichlet Allocation (LDA) baseline. We further improve our results by clustering tweets before classifying.

2. We introduce the idea of topic alignment for topic model evaluation by comparing the latent topics of one model with the outcome of another model as reference. We show the performance of our method using various similarity metrics.

3. We introduce a system for topic discovery that uses the idea of topical alignment in microblogs. In particular, we use a system that detects topics generated weekly and not present in previous weeks and also topics that were only present in previous week and thus died out.
A topic is simply a collection of words that frequently co-occur. One such model is Latent Dirichlet Allocation [5], which allows sets of observations to be explained by unobserved groups that explain why some parts of the data are similar. For example, if observations are words collected into documents, then each document can be seen as a mixture of a small number of topics, such that each word’s creation is attributable to one of the document’s topics.

We believe that clustering of tweets using topic models will help to easily categorize them based on their properties. Using such clusters, we seek to identify the topics or particular event about which the tweet is written. Topic models do not make any assumptions on the ordering of the words in a document and also disregard the grammatical structure. Such a model is also known as the bag-of-words model. This approach is particularly suited to handling irregularities in microblog messages.

Although LDA is a well-known tool for clustering documents based on topics, it does not perform well on microblogs due to the reasons discussed above. Thus, we experimented with two directed graphical models, the Author-Topic (AT) model and the Author-Recipient-Topic (ART) model. The AT model [26] learns topics conditioned on the mixture of authors that composed a document, this has been discussed further in section 3.1. Experimental results show that the state-of-the-art Author-Topic model fails to model hierarchical relationships between entities in social media settings [11]. The ART model [19] is similar to the AT model, but with the crucial enhancement that it conditions the per-message topic distribution jointly on both the authors and
recipients, rather than on individual authors. Thus the discovery of topics in
the ART model is influenced by the social structure in which messages are
sent and received. This setting has been used previously for role discovery in
social networks [19]. We present the ART model for microblogs and analyze
its performance with other models. To the best of our knowledge, our work
is the first time the ART model has been implemented for topic discovery
in microblogs. Our results and analysis have enabled us to make important
inferences about Twitter messages, users and their interactions.

1.4 Publications

During the course of this masters thesis we have the following works
published or under submission:

[1] (Accepted) Nazneen Fatema Rajani, Kate McArdle, Jason Baldridge. 2014. Extracting Topics Based on Authors, Recipients and Content in Microblogs. To appear in proceedings of the 37th Annual ACM SIGIR Conference. Gold Coast, Australia

Chapter 2

Background and Related Work

In this chapter we explain a few background concepts that are necessary to understand this thesis work. We also review some recent research in the broad areas of analyzing online social media using topic models and new topic discovery in microblogs.

2.1 Background

Bishop in his book defines probabilistic generative models for statistical machine learning and natural language processing in the following way. A model that specifies a joint probability distribution over observation and label sequences and can be used for randomly generating observable data given some prior probability distribution [3]. Generative models are used in machine learning for either modeling data directly such as modeling observations drawn from a probability density function or as an intermediate step to forming a conditional probability density function. Baye’s rule can be used to form a conditional distribution from a generative model. N-gram language models, Naïve Baye’s classifiers and topic models are few examples of generative models used in machine learning.
We will briefly discuss these generative models in order to give background to our work:

- **N-gram language models**: A statistical language model assigns a probability to a sequence of $m$ words $P(w_1, \ldots, w_m)$ by means of a probability distribution. In particular, they estimate probability of a word given its prior context, for example, $P($phone$|\text{Please turn off your cell})$. An N-gram model uses only $N-1$ words of prior context. However, the number of parameters required grows exponentially with the number of words in prior context. The Markov assumption is that the future behavior of a dynamical system only depends on its recent history. In particular, in a $k$th-order Markov model, the next state only depends on the $k$ most recent states, therefore an N-gram model is a $N-1$-order Markov model.

- **Naive Baye's classifiers**: They are simple probabilistic classifiers based on applying Bayes' theorem with independence assumptions. A more descriptive term for the underlying probability model would be that the features used in the model are independent. The probability model for a classifier is a conditional model $P(C|F_1, \ldots, F_n)$ over a class variable $C$ conditional on several feature variables $F_1$ through $F_n$. The assumption this classifier makes is that each of the features $F_1 \ldots F_n$ are independent of each other and conditional on this assumption, it estimates the class $C$ which is very often binary.
• **Topic models**: Topic models are generative models and a popular method for modeling term frequency occurrences for documents in a given corpus. The basic idea is to describe a document as a mixture of different topics. A document typically concerns multiple topics in different proportions. For example, in a document that is 90% about fruits and 10% about vegetables, there would probably be about 9 times more words related to fruits than words about vegetables. A topic is simply a bag of words that occur frequently with each other. A topic model captures this intuition in a mathematical framework based on the statistics of the words in each document, what the topics might be and what each document’s balance of topics is.

### 2.1.1 Topic Models and their Characteristics

Latent Dirichlet allocation is a generative model that allows sets of observations to be explained by unobserved groups which explain why some parts of the data are similar [5]. For example, if observations are words collected into documents, it posits that each document is a mixture of a small number of topics and that each word’s creation is attributable to one of the document’s topics. Latent semantic analysis (LSA) is a technique in information retrieval and natural language processing, in particular in distributional semantics for analyzing relationships between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms [15].

Kireyev et al. discussed that topic models have certain properties that
make it suitable to analyze Twitter data [14]. These are summarized below:

Topic models do not make any assumptions about the ordering of words known as bag-of-words model and it disregards grammar as well [26]. This is particularly suitable to our work because we handle Twitter messages that are very unstructured and noisy with regards to language and grammar. Each document is represented as a vector of words that describes its distribution over the topics. This representation is convenient to compute document similarity and perform new topic detection. Training a topic model is easy since it uses unsupervised learning, that is it learns from unannotated data. It saves the effort required to create labeled data and train classifiers that learn on such data. Topic models are useful for identifying unobserved or latent relationships in the data. This makes dealing with abbreviations and misspellings easy by using topic models.

2.2 Related Work

Topic models have been applied to a number of tasks that are relevant to our thesis contribution. We will briefly describe three categories and cite a few examples in each.

2.2.1 Topic Models for Information Discovery

Topic models for information discovery is a well known and extensive area of research but also one with diverse applications. Phan et al. present a framework to build classifiers using both a set of labeled training data and
hidden topics discovered from large scale data collections [21]. They provide a general framework to be applied across different data domains. Steyvers et al. in their 2007 paper present a generative model to discover topics covered by papers in PNAS [25]. These topics were then used to identify relationships between various science disciplines and finding latest trends. Griffiths et al. describe an unsupervised learning algorithm that extracts both the topics expressed in large text collection and models how the authors of the documents use those topics [10]. Such author-topic models can be used to discover topic trends, finding authors who most likely tend to write on certain topics and so on. The Author-Recipient-Topic model is a Bayesian model for social network analysis that discovers topics in discussions conditioned on sender-recipient relationships in the corpus [19].

2.2.2 Topic Models for Text Categorization

Text categorization based on word clustering algorithms was described in [2]. Dhillon and others introduce $k$-means clustering for sparse data [8]. A topic vector based space model for document comparison was introduced and discussed in [1]. Lee et al. explore supervised and unsupervised approaches to detect topic in biomedical text categorization. They describe the Naive Bayes based approach to assign text to predefined topics and perform topic based clustering using unsupervised hierarchical clustering algorithms.
2.2.3 Evaluating Topic Models

Topic model evaluation is another area of research that has gained a lot of attention. Wallach et al. explore the idea of evaluating topic models relative to other topic based models as well as to other non-topic based generative model. The Chib-style estimator and “left-to-right” algorithm presented by them provides a clear methodology for accurately assessing and selecting topic models [28]. Chuang and others develop a framework for scale assessment of topic relevance for domain specific topics. They do this by using the idea of topical alignment between latent topics and reference concepts developed by experts in the domain [7].

2.2.4 Topic Models and Online Social Media

Recent research has started to look at content related aspects of online social media and specifically Twitter. Ramage et al. present use of a partially supervised learning model (Labeled LDA) to characterize Twitter data and users [22]. They classify tweets based on roughly four dimensions such as substance, style, social and status. The topic based clustering approach by Kireyev and others identifies latent patterns like informational and emotional messages in earthquake and tsunami data sets collected from Twitter [14].
Chapter 3

System Design and Implementation

This chapter gives an overview of the design and implementation of our system. It also discusses the methodologies used in our implementation and the pre-processing of our dataset. In the first section we explain the system components that directly affect the system.

3.1 System Design

Figure 3.1 gives a very high level overview of the main components our system.

**Topic Modeler**

Firstly we describe the most important component of our system, the topic modeler. As described in Chapter 1, the job of a topic modeler is to identify and group words that tend to appear together into ‘topics’ based on a probability distribution. Our system uses three types of topic modelers, LDA, AT and ART. We introduce the terminology used and describe each of them briefly. We use the following terminology: a set of documents forms a corpus. The set of unique words that are used in the corpus forms the corpus’s
vocabulary, while we refer to the collection of words that appear in a given document as word tokens. The word tokens found in a document are not necessarily unique words from the vocabulary. For example, a tweet that appears as “twinkle twinkle little star” uses the following words from the vocabulary: twinkle, little, star. The word tokens in this tweet are: twinkle, twinkle, little, star.

**Latent Dirichlet Allocation:** Latent Dirichlet Allocation was first introduced by Blei et al. [5] and provided a probabilistic foundation for Latent Semantic Analysis, improving on LSA. LDA models each text document in a corpus as a mixture of an underlying set of topics. Figure 3.2 displays a graphical representation of LDA. Each document $d$ has a multinomial distribution $\theta_d$ of topics, and each topic $z$ has a multinomial distribution $\phi_z$ of words. A document’s topic distribution is randomly sampled from a Dirichlet distribution with hyper-parameter $\alpha$, and each topic’s word distribution is randomly sampled from a Dirichlet distribu-
tion with hyper-parameter $\beta$. Thus, topic assignment in LDA is modeled solely on the document’s word token content.

**Author-Topic Model:** The Author-Topic model [26] builds on LDA, by modeling a document’s topics based on the document’s content, as in LDA, and by conditioning on the document’s authors. Figure 3.2 displays a graphical representation of the AT model. Each document $d$ has a set of observed authors $a_d$. A document’s topic distribution is influenced by this set of authors. To generate each word token in the document, an author $x$ is randomly and uniformly sampled from $a_d$, and then a topic $z$ is sampled from the author’s topic distribution $\theta_x$, which comes from a Dirichlet distribution with hyper-parameter $\alpha$. From this topic, the word token is sampled from the topic’s word distribution.
φ₂, which comes from a Dirichlet distribution with hyper-parameter β. Thus, topic assignment in the AT model is based on the document’s authors and word token content. A useful application of the AT model is predicting co-authors of a document, book or paper [23].

**Author-Recipient-Topic Model:** The Author-Recipient-Topic model [19] builds on LDA and AT, by modeling a document’s topics based on the document’s content, as in LDA, the document’s authors, as in AT, and the document’s recipients. Thus, ART is only appropriate for documents with specific recipients (e.g., emails) and is not appropriate for documents without recipients (e.g., scholarly articles). Figure 3.2 displays a graphical representation of ART. Each document d has a set of authors a_d and a set of recipients r_d. A document’s topic distribution is influenced by the set of observed author-recipient pairs. To generate each word token in the document, an author-recipient pair ar is randomly and uniformly sampled from this set, and then a topic z is sampled from the author-recipient pair’s topic distribution θ_{ar}, which comes from a Dirichlet distribution with hyper-parameter α. From this topic, the word token is sampled from the topic’s word distribution φ_z, which comes from a Dirichlet distribution with hyper-parameter β. Thus, topic assignment in the ART model is based on the document’s authors, recipients, and word token content. A useful application of the ART model is role discovery and understanding social links as described by [19].
Our topic modeler has three stages:

1. **Input:** This step involves processing and filtering the raw tweets described in Section 3.2 into a format acceptable by MatLab Topic Modeling Toolbox \(^1\). This step also removes stop words from the corpus before feeding it as an input to the topic modeler.

2. **Training:** To train the LDA and Author-Topic models, we used the Matlab Topic Modeling Toolbox, which uses Gibbs sampling to approximate the inference step of extracting topics, since it cannot be done exactly for LDA and similar models [19]. To train the Author-Recipient-Topic model, we note that the approach is identical to the Author-Topic model, if one considers a document’s author-recipient pair to be its author. Thus, we used the Toolbox’s Author-Topic implementation to perform ART modeling, providing author-recipient pairs instead of authors. This stage also requires setting the hyper-parameters \(\alpha\) and \(\beta\) appropriately and supplying the number of topics \(N\) to be used. \(\alpha\) is the prior on the per-document topic distributions while \(\beta\) is the prior on the per-topic word distribution.

3. **Output:** The output of a topic modeler is a list of topics \(N\) containing top words along with probabilities of them belonging to that particular topic. The distribution of words in each topic helps in making inferences

\(^1\)http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm
about that topic. Figure 3.3 gives an example of the output obtained by running the topic modeler on a set of tweets.

3.2 Dataset

This section discusses the tweet corpus and the various pre-processing steps done on the raw data. All our results and analysis are based on the below mentioned dataset. Our dataset comprises tweets from August to October 2008. We used the Twitter Spritzer to extract an initial set of 160,000 tweets from this time period. We also use a set of 80,520 tweets from February 2009 for some of our weekly analysis and has been explicitly mentioned wherever used.

We now describe the filtering we performed on the set of tweets and the word tokens within each tweet.

3.2.1 Filtering on Tweets

The set of tweets was then filtered in two ways, which we describe here. The first filtering we performed was for @mention. We compare the relative performances of the topic models described in Section 3.1, one of
which is the Author-Recipient-Topic Model. This model requires that every
document have at least one recipient, so we filtered our original dataset to
only keep tweets that include @mention, which is the way in which a tweet
author directs his tweet at a specific user. We then consider the Twitter handle
mentioned in @mention to be the recipient of the tweet. In the case of multiple
@mention inclusions in a single tweet, we consider each Twitter handle listed
as separate recipients. Thus, each document consists of three attributes: the
tweet’s content, the tweet’s author, and a set of one or more recipients. We
call this set of tweets the Recipient Dataset. Figure 3.4 is an image of a famous
Twitter conversation.

The second filtering we performed was for hashtag. Our motivation
for this filtering comes from the paper by Ramage et al., which suggests that
the performance of topic modeling on tweets is generally poor, due to the
inherently short nature of each document: every tweet is restricted to 140
characters [22]. Mehrotra and others show that one approach to overcom-
ing this pitfall is to cluster together tweets that contain the same hashtag [20].
Tweets that contain multiple hashtags belong to multiple clusters and depend-
ing on hashtags they contain, are copied into each of those documents. Each
cluster constitutes one document, and the topic model is trained on this set
of documents. The motivation behind tweet pooling is that individual tweets
are very short and hence treating each tweet as an individual document does
not present adequate term co-occurrence data within documents. Aggregating
tweets which are similar in some sense (in our case semantically) enriches the
Figure 3.4: Example of a conversation on Twitter.
content present in a single document from which the LDA can learn a better
topic model. For completeness, we compare the performance of the topic mod-
els described in Section 3.1 when trained on an unclustered dataset to when
trained on a clustered dataset. In order to make such comparisons, we are
required to remove any tweets from the Recipient Dataset that do not have
at least one hashtag in the tweet’s content. We call the resulting dataset the
Single-Tweet Dataset, as each document consists of a single tweet (whose con-
tents contain at least one hashtag), a single author, and a set of one or more
recipients. The Single-Tweet Dataset consists of 7288 tweets, 1176 unique
authors, and 7830 unique author-recipient pairs.

We create a second dataset such that the tweets in the Single-Tweet
Dataset are clustered into documents by hashtag. In the case of a single
tweet with multiple hashtag inclusions, the tweet is included in the document
corresponding to each hashtag. We call this dataset the Clustered Dataset.
Figure 3.5 gives an example of tweets containing #sxsw clustered under the
hashtag #sxsw. In this dataset, each document consists of a set of one or more
tweets (each tweet of which contains the same hashtag), one or more authors
(such that the number of authors is less than or equal to the number of tweets),
and one or more recipients (such that the number of recipients is greater than
or equal to the number of authors). The Clustered Dataset consists of 2563
documents. The numbers of unique authors and unique author-recipient pairs
are the same as for the Single-Tweet Dataset, since the underlying set of tweets
is the same in both datasets.
3.2.2 Filtering on Word Tokens

On both the Single-Tweet Dataset and the Clustered Dataset, we perform filtering on the word tokens contained in the tweets. First, we remove any URLs in the tweets’ contents. Due to the 140-character limit imposed on tweets, many users use shortened URLs, and thus it is not appropriate to contain URLs in our dataset. Next, for word tokens that have an apostrophe followed by a single character, we remove the apostrophe and following character. This allows us to consider, for example, nouns and their possessive form as the same word in the vocabulary. Next, we remove every word token that is of the form “@mention”, to remove recipients from the tweet’s content. We also get rid of any word tokens numbers and non-alphanumeric symbols. Finally, we also remove stop words or frequently occurring words in tweets that do not say much about the tweet’s content. The list of stop words for tweets is slightly different from the standard English stop words and is based on the language model for tweets. Table 3.1 displays some stop words from the list. Our final Single-Tweet Dataset consists of 13,104 unique words, which form
Table 3.1: List of stop words.

<table>
<thead>
<tr>
<th>Twitter corpus stop words</th>
</tr>
</thead>
<tbody>
<tr>
<td>twitter, twitpic, et, hi, rt, lol, get, the, yourself, without, eg, one...</td>
</tr>
</tbody>
</table>

the vocabulary, and 49,387 word tokens. The final Clustered Dataset consists of 12,963 unique words and 57,873 word tokens.

3.3 Implementation

This section discusses the core implementation of our goal to discover new topics [18] by using a suitable topic modeler based on an appropriate input configuration. To achieve this we need to evaluate the topics obtained as output from the modeler. As mentioned in Chapter 1 this is not easy since the model trains in an unsupervised manner. Thus we use the idea of topic model diagnostics described in [7] by Jason Chuang and others, that assesses domain relevance via topical alignment. They built a repository of domain-concepts (reference topics) using expert judgment and quantify topical alignment between a set of latent topics and a set of reference concepts as follows. A topic resolves to a concept if a one-to-one correspondence exists between the two. A misalignment exists when models produce junk or fused topics or when reference concepts are missing or repeated among the latent topics. They used crowd-sourcing to evaluate topical alignment between latent and reference topics.

We implement the above idea with slight modifications and without any manual intervention in the following manner. Firstly, we evaluate the
performance of AT and ART topic models (latent topics) on microblogs by comparing them to the LDA (reference topics) baseline using topical alignment. This also allows us to make important inferences and analysis about the topic modeler used and characteristics of microblogs as discussed in Chapter 4. Next we use the topical alignment technique for new topic detection and prediction, which is the most important contribution of this thesis. We describe the steps that lead us to topic prediction in sections below.

### 3.3.1 Topical Alignment

Topical alignment is the method of aligning latent topics to reference topics or concepts where every topic is a multinomial distribution over words. The likelihood that a latent topic would match a reference topic is the probability of how similar the latent topic is to a reference topic. The output of a topic modeler is for every word in the document, the probability that it belongs to a particular topic. Table 3.2 displays a sample output produced by a topic modeler for ten random words and the probabilities with which they belong to either *Topic 1* or *Topic 2*. Using these probabilities, we can represent topics in a vector space model. Each dimension corresponds to a separate unigram and a topic vector comprises of every unigram’s probability associated with that topic. Therefore the number of dimensions is equal to the size of vocabulary for the dataset. Once the topics are represented as vectors, various similarity measures can be used to find the nearest match between topics. Cosine similarity is one such metric and is used for calculating
<table>
<thead>
<tr>
<th></th>
<th>Topic 1</th>
<th>Topic 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>bad</td>
<td>0.00101</td>
<td>0.00006</td>
</tr>
<tr>
<td>email</td>
<td>0.00078</td>
<td>0.00029</td>
</tr>
<tr>
<td>everyday</td>
<td>0.00054</td>
<td>0.00001</td>
</tr>
<tr>
<td>commercials</td>
<td>0.00078</td>
<td>0.00006</td>
</tr>
<tr>
<td>advertising</td>
<td>0.00078</td>
<td>0.00005</td>
</tr>
<tr>
<td>batteries</td>
<td>0.00054</td>
<td>0.00054</td>
</tr>
<tr>
<td>patrick</td>
<td>0.00078</td>
<td>0.00006</td>
</tr>
<tr>
<td>embarrassing</td>
<td>0.00054</td>
<td>0.00006</td>
</tr>
<tr>
<td>walk</td>
<td>0.00030</td>
<td>0.00078</td>
</tr>
<tr>
<td>digital</td>
<td>0.00006</td>
<td>0.00030</td>
</tr>
</tbody>
</table>

Table 3.2: Probabilities that a unigram is associated with Topic 1 or Topic 2.

Topic similarities. Equation 3.1 gives the cosine similarity measure between topics $T_1$ and $T_2$. Chuang et al. in [7] introduced another similarity metric to improve upon the cosine similarity called the *Rescaled dot product* defined below.

Given a word probability distribution $X$, the scalar $x_i$ denotes the probability for term $i$ in topic $X$. $\vec{X}$ is a vector consisting of all $x_i$ values in descending order, $\hat{X}$ is a vector of $x_i$ in ascending order then rescaled dot product is

$$
\text{Rescaled dot product} = \frac{P \cdot Q - d_{min}}{d_{max} - d_{min}}
$$

$$
d_{max} = \vec{P} \cdot \vec{Q}
$$

$$
d_{min} = \hat{P} \cdot \hat{Q}
$$

Rescaled dot product definition [7].

$$
sim(T_1, T_2) = \frac{T_1 \cdot T_2}{\|T_1\| \|T_2\|}
$$

The similarity calculation between topics gives us an $m \times n$ matrix of all possible pairings among $m$ reference topics and $n$ latent topics. Each entry
Table 3.3: In a correspondence matrix, each entry $p_{r,l}$ represents the probability that a reference topic $r$ and a latent topic $l$ are equivalent. Misalignment scores measure how much topical alignment deviates from an optimal one-to-one correspondence. Comparing a latent topic to all reference topics, junk and fused scores measure how likely the topic matches exactly zero, or more than one reference topic. Missing and repeated scores measure how likely a reference topic matches exactly zero, or more than one latent topic.

$p_{r,l}$ is treated as an independent Bernoulli random variable that represents the likelihood that a latent topic vector representation would be equivalent to a reference topic vector representation. Each of these entries are independent events. We map similarity scores between latent and reference topic vectors into matching likelihoods using thresholding and is discussed in detail in Chapter 4. A correspondence is considered optimal when every latent topic vector maps one-to-one to a reference topic vector and deviations from optimal arrangement leads to misalignments. We consider 4 types of misalignments for a correspondence matrix as discussed in [7] and shown in Table 3.3.

Let $\hat{P}_l(k)$ be the likelihood that there are exactly $k$ matches after comparing a latent topic $l$ to all $m$ reference topics. Similarly let $\hat{P}_r(k)$ be the likelihood that there are exactly $k$ matches after comparing a reference topic $r$ to all $n$ latent topics. Then the 4 types of misalignments defined by [7] are:

**Junk:** The junk score for a latent topic $l$ is the probability $\hat{P}_l(0)$, that
is the latent topic has no matching reference topic.

**Fused:** The fused score for a latent topic \( l \) is the likelihood \( \sum_{k=2}^{n} \hat{P}_l(k) \), that is the latent topic matches two or more reference topics.

**Missing:** The missing score for a reference topic \( r \) is the probability \( \hat{P}_r(0) \), that is the reference topic has no matching latent topic.

**Repeated:** The repeated score for a reference topic \( r \) is the likelihood \( \sum_{k=2}^{m} \hat{P}_r(k) \), that is the reference topic matches two or more latent topics.

### 3.3.2 New Topic Prediction

Topical alignment as discussed above helps us in identifying misalignments between topics. We use this technique to predict new topics in tweets in the following way. We consider tweets on a weekly basis and thus divide our dataset based on the date each tweet was composed. Consider we have tweets from 3 continuous weeks in our dataset and so all our tweets will fall into one of the 3 buckets depending on when they were composed. Next we use the topic modeler and a suitable configuration to obtain lists of word probability distribution for each topic for each week. We then represent the topics for each week as vectors and compute similarities between them using the metrics described in Section 3.3.1. Topics from week 1 serve as reference topics to latent topics of week 2 and topics from week 2 can be used as reference topics to week 3 latent topics. Then similarity is mapped to likelihoods and we compute misalignments between topics of the week under consideration to those
of the previous week. If topics from week 2 are similar in anyway to topics from week 1 then they will either match one-to-one or be repeated or fused. Topics from week 2 that do not find a match to topics from week 1 or as per our definition are junk maybe considered as new topics for week 2 depending on their likelihoods. Figure 3.6 gives the step-by-step process of new topics prediction in our system.

Blei and Lafferty introduced the idea of “Dynamic Topic Models” to track and analyze evolution of topics of a collection of documents over time [4]. As discussed before the order of the words in a document and the order of the documents in the corpus are irrelevant in the training of the LDA, AT and ART topic models. Whereas in dynamic topic models albeit the order of the words is considered exchangeable, the order of the documents plays a fundamental role. The documents are assumed to be grouped by time slice (e.g.: years) and it is assumed that the documents of each group come from a set of topics that evolved from the set of the previous slice. Thus the hyperparameters distributions $\alpha_{t+1}$ and $\beta t + 1, k$ are generated from $\alpha_t$ and $\beta_{t,k}$ respectively.
The authors of the paper assume every generated topic survives and no topics die out at any time. Our system does not make these assumptions and thus is able to detect topics that die out while simultaneously detecting new topics that were generated. The authors also argue that applying Gibbs sampling to do inference in their model is more difficult than in static models, due to the nonconjugacy of the Gaussian and multinomial distributions. Since the new topic detection system proposed by us uses topical alignment, it does not require the more complicated estimation of parameters.

This technique not only allowed us to predict new topics from tweets but also leads to important inferences based on our analysis which are discussed in detail in Chapter 4. We also analyzed latent topics obtained from a topic modeler vs vector representation of hashtags. Going back to our definition of Clustered Dataset in Section 3.2, consider all the tweets that contain a particular hashtag. After filtering these tweets and getting rid of stop words, the vector representation of that hashtag is the probability distribution of all unigrams that occur in those tweets, as discussed in Section 3.3.1. These hashtag vector representations serve as reference topics to topics obtained from a topic modeler. Computing likelihoods between latent topics and hashtag vectors not only allows us to predict new hashtags that have a tendency of trending but also gives us insight into characteristics of microblogs which we discuss in the next chapter.
Chapter 4

Experimental Analysis and Results

This chapter deals with experimental results obtained by our system under various settings. It also discusses various evaluation techniques to understand how well our system performs on microblogs. Finally this chapter also analyzes the results obtained to make important claims and prove them for microblogs.

4.1 Experimental Results

In this section we evaluate the AT and ART topic models that we proposed for microblogs in two ways, one by using an evaluation metric and second by using topical alignment. We also evaluate topic discovery in microblogs using topical alignment. All results in this chapter are based on the dataset discussed in Section 3.2 unless otherwise stated. $N$ denotes the number of latent topics in a topic model; $\alpha$ and $\beta$ denote topic and term smoothing hyperparameters, respectively.
4.1.1 Results using an Evaluation Metric

We present our results performing topic modeling on the Single-Tweet Dataset and the Clustered Dataset. To train the LDA and Author-Topic models, we used the Matlab Topic Modeling Toolbox,\(^1\) which uses Gibbs sampling to approximate the inference step of extracting topics, since it cannot be done exactly for LDA and similar models [19]. To train the Author-Recipient-Topic model, we note that the approach is identical to the Author-Topic model, if one considers a document’s author-recipient pair to be its author. Thus, we modified the Toolbox’s Author-Topic implementation to perform ART modeling, providing author-recipient pairs instead of authors.

4.1.1.1 Model Settings

For all models, we set the model hyper parameters \(\alpha\) and \(\beta\) to \(\frac{50}{|\text{topics}|}\) and \(\frac{200}{|\text{vocabulary}|}\), respectively. In different experiments that we ran on training the models, we used either 500 or 1000 iterations in Gibbs sampling, and we extracted one of the following numbers of topics: 10, 20, 30, 40, 50, 75, 150, 300, 500. We trained three of the models, LDA, AT, and ART, on both the Single-Tweet Dataset and the Clustered Dataset.

4.1.1.2 Model Evaluation

To evaluate our results, we implemented a function in Matlab to calculate a Pointwise Mutual Information (PMI) score for a trained topic model.

\(^1\)http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm
PMI measures the coherence of the topics that are created by a trained topic model, by determining the statistical independence of two words from the same topic appearing together in the same document [20]. The PMI for a pair of words is

\[ PMI(w_i, w_j) = \log \frac{p(w_i, w_j)}{p(w_i)p(w_j)} \]

In our case, for both the Single-Tweet Dataset and the Clustered Dataset, when calculating PMI we consider each tweet to be a document, so we calculate PMI using empirical probabilities of the Single-Tweet Dataset. The probability of a single word, \( p(w_i) \), is the ratio of the number of tweets that contain word \( w_i \) to the total number of tweets. The probability of a pair of words, \( p(w_i, w_j) \), is the ratio of the number of tweets that contain both words \( w_i \) and \( w_j \) to the total number of tweets.

To calculate PMI for a given model, we used the approach outlined in [20]: for each topic, calculate the PMI of each of the possible word pairs among the ten words with the highest probabilities. The PMI for the given topic is the average of the PMI scores for the word pairs, and the PMI for the given model is the average of the PMI scores for the topics. A higher PMI score indicates better topic coherence, and thus we compare each of the trained models based on their PMI scores.

Table 4.1 displays the top 10 words belonging to topic related to “Austin” for each of the LDA, AT and ART topic models.

Our results are displayed in Table 4.2 and Table 4.3. For each number
of topics, the model and dataset combination with the highest PMI is displayed in bold.

Our results indicate that, as expected, the Clustered Dataset results in better-trained topic models than the Single-Tweet Dataset, regardless of the number of topics. By comparing the results with 500 iterations in Table 4.2 and with 1000 iterations in Table 4.3, we do not see a big difference, indicating that our models are converging by 500 iterations. We compare the relative

<table>
<thead>
<tr>
<th>Model</th>
<th>Dataset</th>
<th>PMI score for the following number of topics:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>LDA</td>
<td>Single-Tweet</td>
<td>0.565</td>
</tr>
<tr>
<td>LDA</td>
<td>Clustered</td>
<td><strong>0.770</strong></td>
</tr>
<tr>
<td>AT</td>
<td>Single-Tweet</td>
<td>0.634</td>
</tr>
<tr>
<td>AT</td>
<td>Clustered</td>
<td>0.712</td>
</tr>
<tr>
<td>ART</td>
<td>Single-Tweet</td>
<td>0.523</td>
</tr>
<tr>
<td>ART</td>
<td>Clustered</td>
<td>0.639</td>
</tr>
</tbody>
</table>

Table 4.2: PMI scores for LDA, AT and ART models trained on the Single-Tweet and Clustered Datasets, with 500 iterations.
Table 4.3: PMI scores for LDA, AT and ART models trained on the Single-Tweet and Clustered Datasets, with 1000 iterations.

<table>
<thead>
<tr>
<th>Model</th>
<th>Dataset</th>
<th>PMI score for the following number of topics:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>10</td>
</tr>
<tr>
<td>LDA</td>
<td>Single-Tweet</td>
<td>0.624</td>
</tr>
<tr>
<td>LDA</td>
<td>Clustered</td>
<td><strong>0.746</strong></td>
</tr>
<tr>
<td>AT</td>
<td>Single-Tweet</td>
<td>0.586</td>
</tr>
<tr>
<td>AT</td>
<td>Clustered</td>
<td>0.646</td>
</tr>
<tr>
<td>ART</td>
<td>Single-Tweet</td>
<td>0.612</td>
</tr>
<tr>
<td>ART</td>
<td>Clustered</td>
<td>0.668</td>
</tr>
</tbody>
</table>

performance of LDA, AT and ART across different numbers of topics, by plotting the normalized PMI scores for each of the number of topics, as shown in Figures 4.1 and 4.2. Here we focus on the models that used 500 iterations. Our results suggest that LDA performs better than the other models on clustered tweets for a small number of topics, while ART performs better than the other models on a mid-range number of topics, and AT performs better than the other models on a higher number of topics.

4.1.2 Results using Topical Alignment

Topical alignment can be used for large-scale assessment of topical relevance [7] without human intervention. Most microblog data available is unlabeled and thus topic models need to be trained in an unsupervised manner. Although there is an advantage in learning without annotated data, the downside is that it becomes difficult to evaluate them against a manually labeled gold standard. Crowdsourcing could be used to evaluate the performance of
Figure 4.1: Normalized PMI scores for LDA, AT and ART on the Single-Tweet Dataset (500 iterations).

Figure 4.2: Normalized PMI scores for LDA, AT, and ART on the Clustered Dataset (500 iterations).
topic modelers, however, it would be even harder if not impossible to evaluate the performance of the crowd. By the very nature of microblogs, it is difficult to understand what the user was trying to convey and may not be the same as the output of crowdsourcing. We thus evaluate the performance of AT and ART models on microblogs using topical alignment by comparing them to the well known LDA baseline.

We consider the topics generated by LDA to be reference topics and those by AT and ART modelers to be latent topics. As discussed in Section 3.3.1 we generate a correspondence matrix between the AT or ART topic vectors and LDA topic vectors using cosine-similarity. We vary the number of topics for each of the topic modelers between 100 to 600, \( N \in [100,600] \) and \( \alpha = 5/N, \beta = 0.25 \). Figure 4.3 visualizes the correspondence matrix thus obtained. The color intensity for the diagonal shows the likelihood of a match to be \( > 0.2 \). The matrix visualization displays that there is a greater chance of matching along the diagonals for both AT and ART topics to LDA topics. We discuss this in detail in Section 4.2.

We convert the similarity score in the correspondence matrix between topics to likelihood by using a threshold of 0.5. Thus if two topic vectors have a similarity score \( \geq 0.5 \) then it is likely that they correspond to the same topic. Using this, we obtain misalignments between topics produced by AT and LDA or ART and LDA topic models respectively, Figure 4.4. As expected, a lot of topics were repeated when the AT or ART models were compared to LDA and the ratio of the number of fused and repeated topics for the ART
Figure 4.3: Correspondence matrix between latent AT or ART topic vectors and reference LDA topic vectors. Color intensity shows the likelihood that reference topics and latent topics are equivalent. The x-axis shows the number of LDA topics and y-axis shows the number of AT topics and ART topics respectively.
Figure 4.4: Topical alignment to LDA reference topics for $N \in [100, 600]$ topics (x-axis) and $\alpha = 5/N$ and $\beta = 0.25$. The y-axis shows the fraction of AT or ART topics that have a single matching (resolved), multiple matching LDA topics (repeated) or are subsumed by one (fused) or multiple fused LDA topics (fused and repeated).

model were higher than the AT model. These observations led us to make useful inferences which are discussed in detail in Section 4.2.

Figure 4.5 shows the percentage of resolved topics for AT and ART models when aligned with the LDA reference topics. The highest percentage of resolved topics for both models is obtained for 100 topics, which means that for a small number of topics, the AT and ART models correspond one-to-one most with the LDA topics. Thus we can infer that for smaller number of topics these models detect similar cluster of unigrams which talk about the same topic. This would mean that there are fewer major topics of discussion on Twitter and most users tend to talk about these topics. Secondly considering LDA as a reference to AT and ART models, the plot strengthens our claim.
that LDA performs better for smaller number of topics because it only models based on the content of the blog.

In order to ensure our system is robust to topic matching, we use the method to estimate and removing topical correspondences that can be attributed to random chance by Chuang and others [7]. We consider the correspondence matrix to be a combination of a noise matrix and a definitive matrix. The definitive matrix has entries 0 or 1 and the noise matrix represents chance probability. We assume that the likelihoods of topics matching are randomly drawn from the definitive matrix \((1 - \gamma)\) of the time and from the noise matrix \(\gamma\) of the time, where \(\gamma \in [0, 1]\). The final step is to estimate the parameter \(\gamma\) for optimal value which represents the amount of matches that can be attributed to noise. The optimization problem is discussed in detail in the supplementary material for [7].

Figure 4.5: Percentage of AT and ART resolved topics to LDA topics, x-axis represents \(N \in [100, 600]\).
4.1.3 Topic Discovery using Topical Alignment

While [7] performs topical alignment between topics produced by AT or LDA models and a manually created set of concepts by experts, the idea of topical alignment between two topic models has never been done before and has been proposed for the first time in this thesis. Such type of topical alignment gives a new direction to the research of topic discovery in microblogs and the results obtained by us is described below.

We start with tweets from February 2009 and divide it into weeks, thus obtaining 4 disjoint corpus of tweets totaling 80,520. Next we perform the filtering and pre-processing for each set as described in Section 3.2. We vary the number of topics between $N \in [100, 600]$ and $\alpha = 5/N, \beta = 0.25$. Then we perform topical alignment by considering week 1 topics as reference to week 2 latent topics and week 2 to be reference to week 3 and so on. Figures 4.6 and 4.7 show the topical misalignments for weekly comparison of AT topics and ART topics respectively. Looking at the plots its not surprising that most topics are repeated for a week when compared to the previous week. However what is more interesting is that if we look at the junk topics for a week, it is the likelihood that there were no matching reference topics found for those topics. This would mean that these topics never occurred before this week and thus may be topics of interest or potential new topics. Also the proportion of junk topics is extremely small compared to other types of misalignments and decreases even further as the number of topics increases. By analyzing these junk topics, we found that new topics related to events or disasters like
conferences, elections, flu trends, etc. could very well be predicted using our technique and topics related to tv-shows that happen on a weekly basis was harder to track.

Using weekly topical alignment both AT and ART models were able to predict topics related to #mom2summit, #sxsw, superbowl, flu etc. There were no topics related to #mom2summit in week 1 of February for these models, however there were few mentions in week 2 and a lot more in week 3 which concurs with the fact that mom2summit conference took place on February 19-21 which falls in the 3rd week. Similarly there were no topics related to #sxsw in 1st and 2nd weeks of February but a lot more in the 3rd week which again corresponds to #sxsw taking place in beginning of March and is closer to week 3. At this point, we use the concept of missing topic
Figure 4.7: Topical alignment for weekly ART topics for $N \in [100, 600]$ topics (x-axis) and $\alpha = 5/N$ and $\beta = 0.25$. The y-axis shows the fraction of a week’s topic that have a single matching (resolved), multiple matching (repeated) to previous week’s topics or are subsumed by one (fused) or multiple fused topics of previous week (fused and repeated).

that is the likelihood that there are no matching topics when a previous week’s topic is compared to all current week’s topics. “Superbowl” and “Gaza” are examples of missing topics. Both these topics completely died in week 3 and thus had no matching when comparing week 2 to week 3. This again concurs with the fact Superbowl took place on February 1 in 2009 and Gaza was under war for three weeks in January 2009. “Flu” is an example of a topic that was repeated for all weeks of February for all models. Our findings and analysis concur with the 2009 Twitter trends as well, Figure 4.8\textsuperscript{2}. Table 4.4 gives a list of top 10 words that we could attribute to topics related to “Gaza” and “mom2summit” respectively.

Table 4.4: Top 10 words from topics that were missing for week 2 and junk for week 3 for AT and ART models respectively.

<table>
<thead>
<tr>
<th>AT Week 2</th>
<th>AT Week 3</th>
<th>ART Week 2</th>
<th>ART Week 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>tipster</td>
<td>mom2summit</td>
<td>ixd09</td>
<td>mom2summit</td>
</tr>
<tr>
<td>whistleblower</td>
<td>utah</td>
<td>sunnies</td>
<td>learn</td>
</tr>
<tr>
<td>feared</td>
<td>alltop</td>
<td>globetrotter</td>
<td>agree</td>
</tr>
<tr>
<td>equipment</td>
<td>taught</td>
<td>exploratorium</td>
<td>awesome</td>
</tr>
<tr>
<td>ucsd</td>
<td>mozilla</td>
<td>fusion</td>
<td>stuff</td>
</tr>
<tr>
<td>pacers4got</td>
<td>tabasco</td>
<td>fortune</td>
<td>things</td>
</tr>
<tr>
<td>investing</td>
<td>shade</td>
<td>imaging</td>
<td>red</td>
</tr>
<tr>
<td>imaging</td>
<td>scotch</td>
<td>israel</td>
<td>episodes</td>
</tr>
<tr>
<td>israel</td>
<td>internet</td>
<td>palestine</td>
<td>microsoft</td>
</tr>
<tr>
<td>gaza</td>
<td>market</td>
<td></td>
<td>ca</td>
</tr>
</tbody>
</table>

Figure 4.8: Twitter trends for 2009.
4.1.4 Topic Model Analysis using Hashtags

In this section we would like to use the idea of topical alignment between hashtags and latent topics obtained from LDA, AT and ART topic models. We vary the number of topics for each of the topic modelers between 100 to 600, \( N \in [100, 600] \) and \( \alpha = 5/N, \beta = 0.25 \). The clustered dataset, which contains at least 10 tweets for each hashtag, is used. This gave us altogether 94 different hashtags and a total of 2820 tweets. The hashtag vector is the unigram probability distribution for tweets containing that hashtag after pre-processing and filtering on stop words. We then compute the similarity between topic vectors obtained from each of LDA, AT, ART models and hashtag vectors. Topical alignment between topics vectors to hashtag vectors using a threshold of 0.2 gives us interesting results, Figures 4.9, 4.10. We claim that although only 11\% [12] of tweets actually contain hashtags, the proportion of the tweets that actually talk about topics related to hashtags is a lot more. Our claims can be substantiated by the proportion of resolved topics in Figures 4.9, 4.10. The ratio of tweets that contain hashtags in our corpus is only 6\%, this may be due to the fact that the idea of hashtags was first introduced in 2009. The least proportion of the resolved topics is 18\% and occurs for 400 LDA topics. Analyzing matches between topics and hashtags for this particular setting which we are least confident about proves our claim. This would also mean that drawing inferences by relying on explicit presence of hashtags would not be very useful, for example [6]. Top words for a topic that resolved to #mom2summit were “houston, follow, friday, mom2summit, people, dcth,
Figure 4.9: Topical alignment hashtag vectors for $N \in [100, 600]$ (x-axis) and $\alpha = 5/N$ and $\beta = 0.25$. The y-axis shows the fraction of latent topics that have a single matching (resolved), multiple matching hashtags (repeated) or are subsumed by one (fused) or multiple fused hashtags (fused and repeated).

“single” and are very similar to the most probable unigrams in #mom2summit, “single, mom, mom2summit, houston, friday”.

Another interesting result is that the highest percentage of resolved topics for the LDA model is 23%, for AT is 41% and for ART is 26%, Figure 4.11. Thus we may be able to conclude that the topics obtained from AT and ART models are clustered well with words occurring together in the same cluster or topics and therefore resolve more to the hashtag vectors.
Figure 4.10: Topical alignment hashtag vectors for $N \in [100, 600]$ (x-axis) and $\alpha = 5/N$ and $\beta = 0.25$. The y-axis shows the fraction of latent topics that have a single matching (resolved), multiple matching hashtags (repeated) or are subsumed by one (fused) or multiple fused hashtags (fused and repeated). The LDA topical alignment on the right side is the same as in Figure 4.9 and is repeated here for comparison with the ART model.

Figure 4.11: Percentage of resolved topics for AT and ART models, x-axis represents $N \in [100, 600]$. 
4.2 Discussion

In this section we discuss the inferences that were made from our analysis on microblogs. We first analyze how $PMI$ as an evaluation metric can be used to evaluate the performance of topic models on clustered and unclustered tweets. Next we analyze how topical alignment helped us in discovering new topics and inferences that could be made based on topic models used.

4.2.1 Using an Evaluation Metric

We presented the performance of the ART topic model for microblogs, which addresses the issues of short text modeling. As expected, our experimental results demonstrate that all three types of model perform better on clustered documents than unclustered documents. Tweets belonging to one cluster tend to represent more coherent topics as shown by [20]. Thus, models trained on longer text yield better results than those trained on short text.

Our experiments demonstrate that, on average, for fewer than 300 topics, the performance of the ART model is the best, followed by LDA, and finally the AT model. The poor performance of AT model was a surprise but it enabled us to make important inferences. Firstly we claim that an average Twitter user tweets about a wide range of topics and these topics have a high distance when compared using a similarity metric, thus implying that they have very little or no overlap. Secondly it it very difficult to distinguish between any two average Twitter users; this inference follows from our first claim. Our results provide evidence to these claims, as the performance of
AT model keeps improving as we increase the number of topics. This means that allowing more topics in the model gives room to cluster authors into more topics and allows us to distinguish between their messages. Another point of contention as discussed in [11] is that the reason may be the “OR” nature of the AT model: a message is either “generated” by the message or by an author.

All our models’ performances improve with the number of topics. However, for 500 topics the performance is worse than those for < 500 topics. We suspect this is mainly due to over-fitting. As we increase the number of topics, we perform really well on some documents but for a new document the models fail to estimate the parameters correctly and thus end up misclassifying. So for our data, 300 is the optimal number of topics for topic modeling.

As shown in Section 4.1, increasing the number of iterations from 500 to 1000 has very little effect on the performance of the models. We also ran our models for 2000 iterations and the results did not vary. Thus, we are assured that all our models converge. Also we find that our models are robust to different random initializations to the Gibbs chains.

4.2.2 Topic Discovery using Topical Alignment

It is hard to manually annotate microblogs and even harder to perform large scale assessment of topical relevance. Thus the idea of using topical alignment for discovering topics in microblogs was introduced. Visualizing the correspondence matrix, figure 4.3 indicates that indeed there is some kind
of correspondence between topic models and thus it is meaningful to draw inferences based on such a topical alignment. As expected, many topics were repeated when the AT or ART models were compared to LDA, figure 4.4. The ratio of the number of fused and repeated topics for the ART model were higher than the AT model which would mean that the ART model does a better job of distributing unigrams to the topic they belong to as compared to the AT model. Thus when an LDA topic is compared to ART topics, we find multiple matches which meant that the LDA topic had subsumed those matching ART topics.

At this point we would like to mention that the dynamic topic models (DTM) proposed by Blei and Lafferty do a similar analysis of evolution of topic models over time. They assume that the order of documents in a corpus is relevant, unlike the static topic models. They are able to compare the performance of dynamic and static topic models in predicting topics over time. We on the other hand used the static topic models and the idea of topical alignment to achieve the same end. We haven’t evaluated the performance of DTM on our corpus and we plan to look into it in order to compare it to our system’s performance as part of future work.

We established a few things by using topical alignment for microblogs and analyzing the results. Firstly it helped us in evaluating the performance of AT and ART models on microblogs. Thus we can now say that although LDA works reasonably well on microblogs, the quality of topics generated by the AT and ART models are slightly better because they do a better job
of classifying unigrams to topics. Secondly by computing similarity between weekly tweets we were able to discover new topics and also infer things about topics that are repeated from the previous week or died completely in the current week. Our inferences were based on manually assessing the topics with these characteristics. Finally comparing our topics to hashtag unigrams led us to conclude that although many tweets do not explicitly contain hashtags, the proportion of tweets that do actually talk about these hashtags is a lot higher. Another thing to remember is that this assessment is based on 2009 data when hashtags were not used much.
Chapter 5

Conclusion and Future Work

5.1 Conclusion

The main contributions to the thesis can be summarized as follows. First we demonstrated the application of three types of graphical models, the LDA, the AT and the ART to microblogs. Second we proposed the idea of assessing relevance and topic model performance by measuring misalignments between these models. Third we introduced the idea of topic discovery in microblogs using topical alignment of time-series data. We also introduced the idea of using junk and missing topics for detecting new topics or topics that died out when comparing topics on a temporal basis. Finally these ideas led us to make substantial inferences related to topic modeling in microblogs.

The idea of topical alignment is not new and has been proposed by Chuang et al. in [7]. They used human experts to build a set of reference concepts in the domain of Information Visualization and matched the outcome of various topic models under different settings, to these concepts. They visualize the results they obtain and analyze them in their paper. Although we borrow the idea of topical alignment, we modify it that lead to several unique contributions to this thesis with respect to topic diagnostics. We have no human
intervention and match one topic model against another. We proved that this is meaningful thing to do. We apply the idea to microblogs which have a very different language model compared to larger documents belonging to the same domain. We propose the idea of temporal analysis by matching a model on tweets of different times and thus perform topic detection using topical alignment. Lastly we also match topic models outcomes to hashtag vectors which has never been done before. This allows us to strengthen claims on structure of microblogs.

Further we addressed the issue of topic modeling in a microblogging environment. More specifically, through our experiments we showed that for short and unstructured text, it is more meaningful to cluster the documents before modeling them, leading to superior performance. Our results show that discovering topics by conditioning on the author-recipient relationships in a corpus of tweets works best. In this thesis, we conducted extensive quantitative experiments on the three models. We compared the models based on a number of aspects including how the topics learned by these models differ quantitatively.

We believe that topical alignment would give a new direction to the research on topic discovery in microblogs and that its applications are not limited to large scale assessment of topical relevance for which it was originally introduced. We also believe that although it is good to have a manually annotated set of concepts, it is not imperative to have it for performing topical relevance. Misalignments between topics also help in substantiating claims
that were previously established on characteristics of microblogs. We believe that our research would lay the groundwork for future work in story or event detection in microblogs by implementing topical alignment and its various flavors.

5.2 Future Work

As part of our future work we plan to conduct a more thorough study on evaluation of using topical alignment in microblogs on a temporal basis. We would like to experiment with daily and even hourly tweets to verify robustness of our system. We would also like to conduct experiments in real time. In addition we would like to compare the performance of Dynamic Topic Models (DTM) and our system for topic quality, scalability and also efficiency. Finally we would like to explore the idea of topical alignment for more applications to microblogs.
Bibliography


