MODEL-BASED FEATURE CONSTRUCTION AND TEXT REPRESENTATION FOR SOCIAL MEDIA ANALYSIS

BY

ALEX MORALES

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2020

Urbana, Illinois

Doctoral Committee:

Professor ChengXiang Zhai, Chair
Professor Jiawei Han
Associate Professor Julia Hockenmaier
Professor Lyle Ungar

ABSTRACT

Text representation is at the foundation of most text-based applications. Surface features are insufficient for many tasks and therefore constructing powerful discriminative features in a general way is an open challenge. Current approaches use deep neural networks to bypass feature construction. While deep learning can learn sophisticated representations from text, it requires a lot of training data, which might not be readily available, and the derived features are not necessarily interpretable. In this work, we explore a novel paradigm, model-based feature construction (MBFC), that allows us to construct semantic features that can potentially improve many applications. In brief, MBFC uses human knowledge and expertise as well as big data to guide the design of models that enhance predictive modeling and support the data mining process by extracting useful knowledge, which in turn can be used as features for downstream prediction tasks. In this dissertation, we show how this paradigm can be applied to several tasks of social media analysis. We explore how MBFC can be used to solve the problem of target misalignment for prediction, where the output variable and the data may be at different levels of resolution and the goal is to construct features which can bridge this gap. The MBFC method allows us to use additional related data, e.g. associated context, to facilitate semantic analysis and feature construction.

In this dissertation, we focus on a subset of problems which social media data, in particular text data, can be leveraged to construct useful representations for prediction. We explore several kinds of user generated content in social media data such as review data for useful review prediction, micro-blogging data for urgent health-based prediction tasks and discussion forum data for expert prediction. First, we propose a background mixture model to capture incongruity features in text, and use these features for humor detection in restaurant reviews. Second, we propose a source reliability feature representation method for trustworthy comment identification that incorporates user aspect expertise when modeling fine-grained reliabilities in an online discussion forum. And finally, we propose multi-view attribute features that adapt MBFC to handle the target misalignment problem for topic-based features and apply this to tweets in order to forecast new diagnosis rates for sexually transmitted infections.

ACKNOWLEDGMENTS

Throughout the writing of this dissertation I have received a great deal of support and assistance.

I would first like to thank Professor ChengXiang Zhai for all his patience and his kind-hearted approach to mentoring. His advice was indispensable and instrumental for the completion of this dissertation. Thanks for always encouraging me to pursue my research interests and ideas, no matter how much I doubted my abilities.

I would like to acknowledge my committee members Associate Professor Julia Hockenmaier, Professor Jiawei Han, and Professor Lyle Ungar for all their feedback and comments on my work which has helped me think about my research in a larger context.

Special thanks to the people at National Consortium for Graduate Degrees for Minorities in Engineering and Science, the Support for Under-Represented Groups in Engineering Fellowship Program, the Graduate College, and Professor John Hart for not only taking a chance on me by funding my Ph.D. program but also providing me with opportunities to volunteer and give back to help underrepresented groups.

I wish to thank all of my collaborators, Kanika Narang, Professor Hari Sundaram, Nupoor Gandhi, Professor Bo Li, Sophie Lohmann and everyone in the Social Action Lab. I warmly express gratitude to Ismini Lourentzou, Assistant Professor Sally Chan, and Professor Dolores Albarracin, who provided mentorship and guidance throughout my studies. Moreover, I thank all Text Information Management and Analysis group members and peers for their fruitful, and some not so fruitful, discussions.

On another note, I would like to thank close friends Lauren and Colin for the happy distractions, fond memories, and for carrying our trivia team. I am also grateful for Anjali's companionship and support on this journey together.

I express my deepest gratitude to my family for keeping me grounded when I needed to be. I am indebted to my parents for instilling in me hard work and perseverance and shaping my character.

TABLE OF CONTENTS

CHAPTER 1: INTRODUCTION

The proliferation of user generated content on the web has enabled many applications ranging from text prediction and personalized recommendation to automatic dialogue generation. In a social media context, text data broadly refers to data generated online, such as emails, blogs, micro-blogs, reviews, questions and comments in online forums, and text chat messages. In social networks, this can either be in a social setting, where the users communicate with each other, or using more traditional methods, where interactions are instead more broadcasted. Thus, we can view users in a social media context as live sensors and observe this transfer of information via their text footprint. A broad goal of this dissertation is to understand user behavior by modeling different aspects from the text footprint in a social media setting. For example, we encode our understanding of the user behavior via the generative process of how a user writes a funny review, and how the user's expertise is modeled across various aspects. In doing so, we can encode our understanding of the user's behavior and construct features from these models that could be reused and potentially be applied to many prediction problems.

Big data offers a great opportunity for applying prediction algorithms on text data in order to extract and discover useful insights for decision making. The rise of deep learning algorithms and methods has enabled many useful applications involving big text data [1]. Such applications in the banking and finance industries include credit card fraud detection [2], while applications in the health sector include using social media for epidemic prediction [3] and outbreak detection [4] and have helped shape policies in governance [5]. However, deep learning methods lack interpretability and model transparency, which make them difficult to apply to situations where there is a need for fairness and accountability in decision making [6]. This need for interpretability in decision making, such as in text prediction, motivates the need for model-based feature representation and construction.

In machine learning and pattern recognition, a feature is an individual measurable property or characteristic of a phenomenon being observed [7]. The feature construction process can be designed to encode human intuition. It is often the case where machine learning practitioners rely on experience and intuition to determine which features are the best suited for the problem at hand. For example, in citation recommendation, where the goal is to recommend reference articles for a given scientific article, there are many features that can be considered, such as the article's title, authors, venue, content text, and the context in which the citation occurs [8]. In general, feature construction is challenging since there is no clear indication of which features to use, when to use those features or how to use them [9].

There is also little notion of feature re-usability: the features constructed for one problem may not be general enough to work for other similar prediction problems.

In this dissertation, we study the problem of text-based feature representation for text classification systems. We describe a framework for reasoning about model-based text features derived from data mining methods which encode human expertise and domain knowledge. We tackle the challenge of feature construction in social media by considering how different types of associated data, or context, can be used to enhance features. First, we consider the companion text that can be derived from the entities mentioned in the text. In particular, we propose a probabilistic mixture model to encode background text sources and show how this is useful in identifying incongruity features in humorous user reviews. This model-based feature representation is an instance of *differential semantic feature representations*, a class of feature representation that relates the meaning of a source text to a reference dataset. Second, we consider the user, or source, of the text and how we can model not only what they have written, but also to what extent can the text be trusted. We propose a method to better analyze user reliability by modeling fine-grained reliability distributed over several aspects in the context of comment trustworthiness discovery. Third, we consider the meta-data associated with social media posts to repurpose model-based feature construction methods. We propose a multi-view of feature construction which can encompass associated meta-data such as location, time, authors and social media-specific attributes. While the problem of feature representation is a general one, we focus on social media as a case study for the model-induced feature construction methods. We explore several datasets and communities such as restaurant review data from Yelp, micro-blogging data from Twitter, and online discussion forum data from Reddit. We ground our models in a variety of applications to show the performance improvements of using model-based features over traditional and classical features.

1.1 TEXT REPRESENTATION AND FEATURE CONSTRUCTION

In a text prediction task, a machine learning algorithm is used in conjunction with the feature representation of the input text to produce some outcome variable. The purpose of feature and text representation is to create a meaningful machine-readable representation of the input data. We can conceive countless ways of creating these representations using our domain knowledge and intuition; however, many representations may be irrelevant to the problem at hand, or may not necessarily generalize, and scale, to similar problems. An alternative line of research forgoes part of the feature engineering and has the learning model, typically a neural network, learn such representations. While both methods have

their advantages and disadvantages, they may not necessarily be mutually exclusive: we can leverage some model-learned representations to design novel features, and we can use domain knowledge to refine automatic learning of such representations.

Text representation and feature construction have been long-standing and important facets for many domains, such as information retrieval, machine learning and natural language processing [10]. Text representation with respect to text prediction and text mining has changed much since its inception. In the information retrieval (IR) community, the problem of feature representation of text has been well studied [11] and has resulted in a wealth of feature construction methods. Some examples include bag-of-words (BOW) representations, statistical phrase indexing [12], syntactic phrase indexing [12, 13, 14] and latent semantic indexing [15, 16], as well as important weighing heuristics for document retrieval and document-query weighing tasks [17, 18, 19].

In this section, we provide a brief literature review of the major works for feature construction. We first focus on shallow text-based features, but also show that these features can be augmented to include other forms of related data. We then describe semantic features, and, finally, introduce model-based features.

1.1.1 Lexical and Shallow Features

To represent a basic lexical unit of text, namely words, there have been considerable efforts in developing features which capture different categorical discrepancies. Perhaps the most standard of features to use in text classification, bag-of-words features only use the most basic of lexical units, the words in each instance, to construct the representation. The simplicity of BOW features therefore make them applicable to many use cases. However, BOW features are limited by the fact that they are not able to capture lexical ambiguity. Another frequently used feature type in IR and natural language processing (NLP) tasks is part-of-speech (POS) tags. The feature measure here maps each word $w \in \mathcal{V}$ to a POS tag set [20, 21]. In both cases, these shallow and lexical features fail to capture word semantics.

Another issue with using words as features is that it is possible to observe new unseen words when applying these features to new data. To overcome this limitation, one approach is to label these words with an explicit "UNK" label. However, by using this approach, we may lose some information about the unknown words, which may be lexically similar, e.g. misspelled or plural forms, or semantically similar. To combat this, alternative tokenization strategies have been proposed, such as byte-pair encoding [22], which breaks words down into subwords, and unigram tokenization [23, 24, 25], which does so in a morphologically meaningful way. An alternative approach is to use character n-grams to represent words

at the character level. An n-gram is a contiguous set of n items; this can include words, subwords, or characters (for a more detailed overview, see [26, 27]). These approaches have been shown to capture morphology, author style and syntactic information [28] and have been shown to work better than word n-grams in some tasks, such as hate speech and abusive language detection [29, 30, 31].

1.1.2 Probabilistic Topic Modeling and Semantic Features

Feature representations that encode word semantics partially overcome some limitations of shallow lexical features. Some of the notable works of creating semantic feature representation of text include latent semantic indexing [15] and topic modeling [16, 32].

The idea for latent semantic indexing approaches is to project the term-frequency matrix to a lower dimensional semantic space. These projections can be performed via rank k approximation of singular value decomposition or non-negative matrix factorization. The key is that these projections retain the word semantics, and thus produce a better approximation for the document similarity than the BOW representation.

If the projections are performed in a non-negative matrix factorization, the resulting representations can be interpreted as topic distributions [16]. Topic modeling approaches have been successfully applied in a variety of tasks and have been developed for different purposes [33, 34, 35, 36]. Here, we focus our discussion on Latent Dirichlet Allocation (LDA) [32], since it is one of the most popular and widely-used methods for topic modeling. LDA has also been extended to many variants that handle different types of data and assumptions [36, 37]. LDA can be described via a generative graphical model, which describes a process for generating each word in a document by first sampling a topic distribution for the document and then sampling a word from a single topic. LDA can capture the co-occurrence patterns of terms across document in a corpus. LDA is a powerful text mining method which can discover various topics, or themes, across documents and cluster similar documents via these themes.

Features generated using LDA are two-fold: (1) the document-topic distribution is a topic feature representation of each document, and (2) the topic-word distribution is a corpus-wide summary describing the topics. Similar to LDA, a focus of our work is to develop text mining models with a purpose, such that they can mine useful and interpretable knowledge while also being useful for feature construction and text prediction.

1.1.3 Word Representation Features

In recent years, with the rise of deep learning methods, text representation has been largely left to neural network models. The key distinction here is that the lexical units, i.e. words, are embedded in a high-dimensional vector space [38]. Early approaches to constructing word embeddings, such as word2vec [39] and GloVe [40], assumed a static representation for each word. In contrast, current state-of-the-art approaches consider the context of the word when building its representation. These approaches can thus capture the polysemy aspect of words [38, 41]. While the discriminative power of these deep learning approaches has been observed through various applications, one major limitation is the lack of faithful interpretability [42, 43].

1.2 THESIS CONTRIBUTIONS

In this dissertation, we explore a novel strategy that we call model-based feature construction (MBFC), that emphasizes the use of domain knowledge as well as companion text data in social media to construct features. In particular, we address the question of reusability in the feature construction process by providing a multi-view attribute approach for feature construction. Like LDA, MBFC provides a way to construct discriminate features for prediction tasks via a model for knowledge discovery. We analyze the benefit of applying MBFC to three different social media prediction tasks, namely recognizing humor in reviews, judging comment trustworthiness on online discussion forums, and predicting new STI diagnoses using tweets. In each of these tasks, we leverage different types of context found in social media text; these include background reference text, user interaction networks, and associated meta-data, respectively. In doing so, we show that MBFC is a general framework for discriminative feature construction for social media text.

1.2.1 Social Media Analysis

The growth of online feedback systems, such as reviews, discussion forums, and blogs in which users can write about their preferences and opinions, has enabled more creativity in the written communication of user ideas. As such, these feedback systems have become ubiquitous, and it is not difficult to imagine a future with smart systems reacting to user's behavior in a human-like manner [44].

Social media networks and online discussion forums have been shown to provide valuable insights; examples of discoverable insights include opposing discussion opinions [45], user

metrics, and adverse drug effects [46]. Additionally, online reviews for services and products have been shown to have a real monetary impact on revenue [47].

Social micro-blogging sites like Twitter have enabled the democratization of disruptive communication for social activism and have played a key role in social movement organization such as #MeToo, Arab Spring, Black Lives Matter, and the Occupy movement [48, 49, 50, 51, 52]. While the user generated content may be useful, most social media platforms have almost no regulations on post requirements or user background; as a result, many responses contain low-quality, conflicting, and unreliable information [53]. This misinformation could lead to severe consequences, especially in health-related forums, that outweigh the positive benefits of these communities. To address this challenge, some forums employ moderators to curate appropriate responses; however, it is not only expensive to curate each reply manually, but also unsustainable [54].

In this dissertation, we explore several applications that leverage social media data. First, to show the utility of modeling incongruity in text, we leverage Yelp reviews as a form of creative user-generated text and use them in conjunction with Wikipedia to develop features for humor identification [55]. We then use our source aspect-reliability features to identify comment trustworthiness in community discussion forums. To do this, we create the CrowdQM Reddit dataset, in which users across three different Reddit communities seek help by posting questions and get answers from other users in the community by the replies or comments; we then show that our model-based features can be used to predict expertise in the community [54]. In the last line of work, we show that we can use multi-view attribute features in conjunction with social media data and meta-data to predict health-related markers. We use Twitter data to predict the new diagnosis incident rates of four sexually transmitted infections and diseases (STIs) and provide some guidance on how to use model-based features to improve STI rate forecasting [56].

1.2.2 Model-Based Feature Construction

In this dissertation, we study the notion of model-based features on text prediction tasks by considering various contexts of text data. Since in prediction tasks, such as text classification and text regression modeling, we can directly observe the discriminative power of features, this setting is more appropriate than other tasks, such as clustering, where these features could potentially also be applied. The goal of model-based features is to allow the inclusion of conjectures of human expertise while mining useful knowledge.

Model-based feature construction is an unsupervised method for text mining that models some context of the data to produce a low-dimensional feature representation. In some

cases, this representation can be used as a synthesized version of the text data, or as a representation of user attributes such as reliability, which make it suitable for different prediction tasks. Specifically, model-based features assume there are some latent variables of the text which can be encoded via an interpretable latent-variable model. This is not solely limited to text, but also the agents producing the texts as well.

Compared to lexical features, model-based features can encode a variety of semantic information beyond the text's surface form, such as user aspect-reliability. Unlike deep learning and neural network representations, the model-based features are interpretable; that is, each latent variable is derived in some explainable manner. While topic model-based methods can be regarded as model-based features, they limit the feature construction model to in-domain corpora. However, when we read a piece of text, we typically incorporate background world knowledge from which we can make connections; in some cases, the author assumes this knowledge and makes specific references. We therefore propose a novel way to construct features using models that leverage different, possibly out-of-domain contexts in which text appears, such as reference background text.

We develop three models which capture different contexts of the data. In the first work, humor recognition, we model human background knowledge by appending a reference text corpus by which we can describe the references made by the author. In the second line of work, comment trustworthiness identification, we model each user as a source with a unique range of expertise and capture the user reliabilities via their text footprints. Finally, in STI new diagnosis prediction, we leverage social media meta-data, such as user location and hashtags, to model different contexts of the text data.

Specifically, we propose and study the following novel strategies for developing model-based features for text prediction.

Semantic Incoherence Features via Background Text An aspect of feature construction is the usage of external text. Typically, there is auxiliary text which can augment the task-specific data, we call this *background text*. For example, Yelp users write restaurant reviews, but may not solely be limited by a review format. Instead, they may incorporate creative writing, which makes it difficult to process the text [57]. Thus, we would need to leverage background text to understand some common-sense knowledge the users may reference.

We provide models for text along with different contexts, considering reference corpus we develop differential semantic feature representations. Differential semantic features can capture semantic differences of a source text when compared to some background text. For some tasks (e.g., humor detection) it is useful to measure semantic cohesion of text or rather

in coherence of text.

Specifically, we study the problem of automatically identifying humorous text from a new kind of text data, i.e., online reviews. We propose a generative mixture model, based on the theory of incongruity, to model humorous text, which allows us to leverage background text sources, such as Wikipedia entry descriptions, and enables construction of multiple features for identifying humorous reviews.

Evaluation of these features using supervised learning for classifying reviews into humorous and non-humorous reviews shows that the features constructed based on the proposed generative model are much more effective than the major features proposed in the existing literature, allowing us to achieve almost 86% accuracy. These humorous review predictions can also supply good indicators for identifying helpful reviews.

Measuring Multi-Aspect Source Reliability Social media data not only contains the text footprint of the users in the social network, but also the interaction between the users in the network. By studying the user behavior and analysing user relation patterns we can then incorporate this information in the feature construction. By developing *context-based features*, these feature representations can encode information about the domain as well as the task specific problem. Feature construction and thus word representation may be learned by leveraging the social network.

In social media data, text data rarely occurs in isolation; considering source of texts we develop multi-aspect source feature representations. Multi-aspect features are analogous to topic features in the case for sources, as they can capture co-occurrence patterns as well as the source's text.

Community discussion forums are increasingly used to seek advice; however, they often contain conflicting and unreliable information. Truth discovery models estimate source reliability and infer information trustworthiness simultaneously in a mutual reinforcement manner and can be used to distinguish trustworthy comments with no supervision. However, they do not capture the diversity of word expressions and learn a single reliability score for the user. CrowdQM addresses these limitations by modeling the fine-grained aspect-level reliability of users and incorporate semantic similarity between words to learn a latent trustworthy comment embedding. We apply our latent trustworthy comment for comment ranking for three diverse communities in Reddit and show consistent improvement over non-aspect-based approaches. We also show qualitative results on learned reliability scores and word embeddings by our model.

Multi-view Model-Based Aspect Features by leveraging Context Networks In this line of work, we propose a general method to identify the best way to apply an existing model for feature construction to the new problem at hand.

A particular problem for social media data for prediction problems, is there is no one-fit-all solution for the representation and usage of documents. For example, on Twitter messages are in the form of tweets, while the predictive outcomes may be location specific, e.g., the influenza rate for a particular county. Thus, there is a form of mismatch for the observed data and the target variable, we call this, *target misalignment*. We show a method for document representation which tries to solve this problem.

Different than the previous two works, multi-view aspect features address the question of how to apply model-based features in a social media setting by leveraging meta-data information. We develop multi-view features and show that model-based features can benefit indirect prediction task, where there is text misalignment.

The effectiveness of social media-based prediction highly depends on whether we can construct effective content-based features based on social media text data. Features constructed based on topics learned using a topic model are very attractive due to their expressiveness in semantic representation and accommodation of inexact matching of semantically related words. We develop a novel general framework for constructing multi-attribute topic features using multi-views of the text data defined according to metadata attributes and study their effectiveness for a text-based prediction task. Furthermore, we propose and study multiple weighting strategies to align text-based features and prediction outcomes. We evaluate the proposed method on a Twitter corpus of over 100 million tweets collected over a seven-year period in 2009-2015 to predict human immunodeficiency virus (HIV) new diagnosis and other sexually transmitted infections (STIs) new diagnosis in the United States at the ZIP Code-level and county-level resolutions. The results show that feature representations based on attributes such as authors, locations, and hashtags are generally more effective than the conventional topic feature representation.

1.3 OUTLINE

This dissertation studies model-based feature construction with the following specific social media application as case studies. However, we note that the proposed feature construction methods are mostly general and thus can easily be applied in other application going beyond social media.

The remainder of the dissertation is organized as follows. We discus related work and provide an overview of model-based feature construction in Chapter 2, and give a brief

analysis of model-based features in comparison to related work. In Chapter 3 we discuss differential semantic features which incorporate reference background text, in the context of incongruity features for humor identification. Chapter 4 considers multi-aspect source reliability features for community discussion forums, using an optimization framework to incorporate user-behavior and comment posting patterns. In Chapter 5 we consider the problem of constructing features by leveraging network context (or meta-data). Finally, we provide a summary of model-based feature construction and conclude in Chapter 6.

CHAPTER 2: MODEL BASED FEATURE CONSTRUCTION

In this chapter we provide an overview of the related work in feature construction. First, we provide an overview of textual features typically used in social media tasks. Since there are many tasks on social media, and a diverse set of features for those tasks, we organize the review-based task specific vocabulary approaches, semantic and content features, and finally based on word representation learning. The second part of the literature review focuses on works more closely associated with the proposed work in this dissertation. Specifically, we focus on features which use reference corpus and source feature representations in the context of social media analysis.

2.1 BACKGROUND AND PRELIMINARIES

In this dissertation we study the feature representation problem and explore different aspects which make this problem challenging. Consider the following process for generating features, given some measure, m_i, of text, we can create a feature measure for a text unit $d \in E$ as follows:

$$f_i = m_i(d) \tag{2.1}$$

The *feature measure* takes any piece of text and maps it to a real valued representation, e.g., scalar, vector, or tensor. Note that in equation 2.1, m_i can be any feature of text, such as number of unique words (e.g., word length), or it can be rule-based such "1" if the presence of a particular word ω_i in some vocabulary $\omega_i \in \mathcal{V}$ and d can be any piece of text, see Figure 2.1a. We define features which can be measured at the surface-level of text as *direct features*, given by equation 2.1. These features include lexical, word counts and n-gram like features, since we can directly measure these types of features solely from the text document. Alternatively, features that require information beyond the text document we define as *indirect features*, given by equation 2.2.

$$f_i = m_i(d, \mathcal{D}) \tag{2.2}$$

Note that in equation 2.2, we call $\mathcal{D}$ the context used to generate the feature f_i. For example, $\mathcal{D}$ can be the training corpus, if we are generating corpus-level statistics for each feature. In this dissertation we explore several ways we can define this context such as an external reference corpus (Chapter 3), data associated with user-level information (Chapter 4), or an associated meta-data corpus (Chapter 5).

While d is traditionally used to represent a document in information retrieval, we distinguish the text instance for prediction and a document. For example, in the case of predicting stock prices using news articles, the text instance may be the collection of corresponding news articles for the specific company, while a document is a single news article. The news articles can then be used to measure signals about the company's stock price to construct indirect features.

2.1.1 The Feature Construction Process

In this dissertation we make a distinction between the feature construction process from feature weighting, and feature selection methods as used in text prediction problems [58]. These processes can be described as operations on top of a feature measure to further refine the feature measure and improve performance, such as re-normalizing and sub-sampling from the original features. However, selecting what feature weighting scheme and feature sampling scheme to use, and when, are both prominent areas of research [58, 59].

The feature construction process for a text prediction task, stems from the following two questions

- *What signals are useful for the task?*
- *How can we measure these signals as features?*

The first question depends on both the text prediction task and the available data for the task. Identifying which features are the best for the prediction task is largely left to discovery and exploration, as there is one-fit-all approach. In the case of text classification, we may rely on cues from language to identify discrepancies between different classes. Some useful signals might be words, phrases, punctuation, taxonomies or ontologies of features [60, 61], domain specific lexicons and other lexical features [62, 63]. However, these direct features may fail to capture useful signals which indirect features maybe be more suitable for, i.e., semantics or some common-sense knowledge about the world, such as background named entity representation, word representation, author topical expertise, and other background context information.

The second question relates to how we can measure the signals proposed in the first question, e.g., how to construct measures of text data that encapsulate the signals proposed. In general, there are many ways to construct these measures, popular methods include vector space representation or mode, graph representation of text, and embedded features [64]. In our work, we propose MBFC as a method for developing features that capture different

context of social media and in doing so, we propose three different algorithms for deriving features from models. Another challenge pertaining to this question is when the signals which we think might be useful are not at the same granularity as the target variable, we call this the *target misalignment problem*. For example, coming back to our stocks example, if we believe that newsworthy events, e.g., those reported in relevant news sources, may influence stock change, then we can capture these signals via the text in the news articles. The target misalignment problem is a different kind of problem for feature construction, since we can have a good feature measure of text, but if we cannot align this feature to the granularity of the target, it is unusable [56].

The role of text representation and feature construction has been a long-standing important facet for many domains, such as information retrieval, machine learning and natural language processing [10]. In the remainder of this section, we provide a brief literature review of the major works for feature construction. We first focus on shallow text-based features, but also show that these features can be augmented to include other forms of related data as well. We then describe semantic features and introduce model-based features.

2.1.2 Lexical and Shallow Feature Construction

We describe the construction of these features, by providing an example for the problem of trustworthy comment discovery, which we outline in detail in Chapter 4. In brief the goal of trustworthy comment discovery involves selecting the most trustworthy and appropriate comment replies for a submission in an online discussion forum. We defer the discussion on how we can determine the trustworthiness of users and comment and describe here a method for determining relevant comments. Consider the following example in Table 2.1, taken from the CrowdQM Reddit Ask dataset [54]. In Table 2.2 we show the bag-of-word

Submission:	How does AC electricity charge a DC device?
Comment 1:	Alternating current may be converted into direct current using a rectifier.
Comment 2:	The chargers for your battery-powered electronics step down the AC voltage and convert it to DC.
Comment 3:	AC is converted to DC using a "bridge rectifier".

Table 2.1: Example of a submission with corresponding comments. The comments have been shorten for brevity.

feature counts, with minor prepossessing, e.g., eliminating stop words and converting to lower-case. The simplicity of the bag-of-word features makes it applicable to many use

cases, since the only information used is the most basic of lexical units, words in each instance. However, it is limited by the fact that it is not able to capture lexical ambiguity. In the running example, the first comment, C1, does to explicitly mention the term "AC", instead the author uses "alternating current" similarly with "DC". In our example task, for finding the most trustworthy comment, using simply BOW features the first comment will be regarded as not relevant. A limitation of these shallow and lexical feature types is that they fail to capture word semantics. The construction of these feature types can be generalized

vocabulary	S	C1	C2	C3
ac	1	0	1	1
alternating	0	1	0	0
battery	0	0	1	0
convert	0	0	1	0
converted	0	1	0	1
current	0	2	0	0
dc	1	0	1	1
direct	0	1	0	0
does	1	0	0	0
electricity	1	0	0	0
electronics	0	0	1	0
powered	0	0	1	0
rectifier	0	1	0	1
step	0	0	1	0
using	0	1	0	1

Table 2.2: The bag of word representations for the Submission (S) and Comments (C).

by the vector space model in the context of text document representation [65]. BOW feature construction for text prediction can be thought as applying the feature measure function to each dimension in the vector-space model, such that the resulting vector corresponds to the text instance for prediction,

$$\boldsymbol{tf}(d) = \begin{pmatrix} tf_1(d) \\ \vdots \\ tf_{|\mathcal{V}|}(d) \end{pmatrix}. \tag{2.3}$$

In Equation 2.3, $tf_i(d)$ is the term-frequency weight. In general, this can be any measure of the text instance for prediction, such as, the frequency of a character in the text instance, the average word length, the average sentence length, or a measure for the alliteration used in the sentences. In the original vector space model m_i corresponds to a weight scheme for each term

in the corresponding document, such as term-frequency and inverse document frequency (tf-idf). To represent corpus-wide induced feature parameter we can represent $m(d; \mathcal{D})$, where $\mathcal{D}$ is the corresponding corpus for which the feature parameters were estimated,

$$\mathbf{tfidf}(d; \mathcal{D}) = \begin{pmatrix} tf_1(d) \cdot idf_1(d; \mathcal{D}) \\ \vdots \\ tf_{|\mathcal{V}|}(d) \cdot idf_{|\mathcal{V}|}(d; \mathcal{D}) \end{pmatrix}. \tag{2.4}$$

In Equation 2.4, $idf_i(d, \mathcal{D})$ is the estimated inverse document frequency of term $w_i \in \mathcal{V}$. Tf-idf, as a feature, is thus a multi-dimensional feature vector which depends on a corresponding corpus. How to define the feature weights is a prominent area of research with well-known methods such as Okapi weighting, also known as BM25 [66]. BM25 has been extended to handle adaptive term frequencies [17, 18], use structural similarity in queries [67], and extended for co-occurrence graph query expansion methods [68]. For completeness we describe some feature weighting scheme methods, although a thorough examination is out of the scope of this work. In [69], the authors propose a modified version of tf-idf which considers the absent terms in calculating the terms' weights. Other tf-idf weighting schemes take in to account the class discriminative power [70, 71], weighting schemes based on relevance frequency [72], and model induced term weighting schemes [19].

In the context of text prediction, $\mathcal{D}$ is typically set of all examples, however $\mathcal{D}$ is not limited to this set and can be other external or background dataset, we explore this in Chapter 3.

2.1.3 Probabilistic Topic Modeling and Semantic Feature Construction

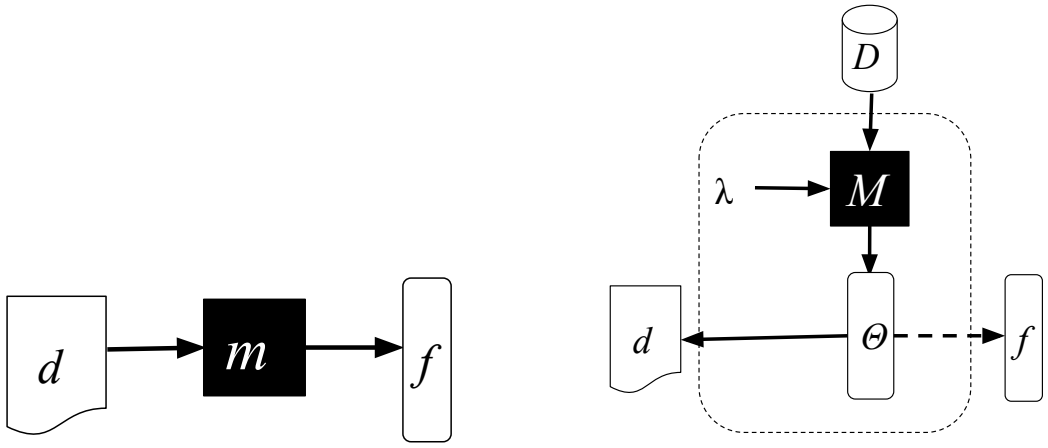
If the projections are performed in a non-negative matrix factorization the resulting representations can be interpreted as topic distributions, [16]. Topic modeling approaches have been successfully applied in a variety of tasks and have been developed for different purposes [33, 34, 35, 36]. Here we limit our discussion to one major work Latent Dirichlet Allocation (LDA) [32]. LDA can be described via a generative graphical model which describes a process for generating each word in a document via first sampling a topic distribution for the document and then sampling a word from a single topic. LDA can capture the co-occurrence patterns of terms across document in a corpus. LDA is a powerful text mining method which can discover various themes, i.e., topics, across documents and cluster similar documents via these themes. We give a more detail account of LDA in chapter 5.

Features generated using LDA are two-fold, first the document-topic distribution is a topic feature representation of each document, and second the topic-word distribution is corpus-

wide summary describing the topics. Treating LDA as a black box model we can describe a feature construction as follows, given a model family $\mathcal{M}$, and a corresponding text collection $\mathcal{D}$, thus,

$$\mathcal{M}_\Lambda(d, \mathcal{D}) = \Theta_\mathcal{D}(d) \tag{2.5}$$

where Λ is the model parameters and $\Theta_\mathcal{D}(d)$ is the estimated features corresponding to the example instance $d \in E$ given by model $\mathcal{M}_\Lambda$, we call these *model-based features*. The feature construction process for model-based features and feature sets is shown in Figure 2.1b.



(a) A view of the feature construction process as a measure of some property.

(b) Model-based features which can depend on external sources to output new parameter estimates that can be used for feature development.

Figure 2.1: Feature construction process for text prediction problems.

In the case of LDA, $\Theta_\mathcal{D}(d) = \theta_d$ is often the topic distribution for d. Note that this equation is similar to equation 2.1, with the notion that our measure is now a model. The focus of this work is then developing text mining models with a purpose, that can mine useful knowledge, and be used for feature construction and text prediction.

2.1.4 Word Representation Features

Word embeddings derived from deep learning models can be categorized as features derived from models as they fit the definition. However, they are not interpretable, so briefly explore how to generate these types of feature representations, we take a closer examination on how

we can leverage these types of features to perform text prediction in Chapter 4.

2.2 FEATURE CONSTRUCTION FOR SOCIAL MEDIA TEXT DATA

We can categorize the feature construction in vocabulary approaches, either manually curated or data-driven, syntactic/lexical based features, semantic-based features and finally representation-based features. Since these features are constructed with some purpose in mind, we also describe the tasks associated with these features, i.e., where the features have had success and are generally useful. Attention is given to health-related and sentiment domains, since they are the most relevant to our work, but also because these growing areas have a large potential for impacting many people's lives.

2.2.1 Closed-Vocabulary Features

Feature construction and development is typically situated in some prediction context. That is, features representations are used with some specific intent, for example in topic categorization, such as identifying tweets related to HIV or other STIs, manual constructed keywords, or closed vocabularies, can be used as features to improve feature representation, [73, 74, 75, 76]. The purpose of a closed-vocabulary approach stems from a prior knowledge or assumption about what may work well for the prediction task. A major line of application is harnessing social media for health information, due to the wealth of information shared, such as people seeking medical advice, posting side effects, and sharing health related content [77]. Seminal work in this domain is in comparing Google Flu trends query data with rates of influenza [78, 79]. Researchers have replicated this analysis on Twitter using a large gazetteer of flu related terms and have shown it is possible to use tweets to predict flu trends [80]. They show that it is possible to monitor influenza activity, to some extent, in the united states by simply counting frequencies of terms such as 'flu' and 'influenza'. However, relying solely on the search queries leads to an overestimation of influenza, namely because there is no distinction between general awareness about the flu and searches for treatment methods [81, 82]. The estimation error may also be compounded by the fact that the language in social media tends to be noisy, since there are many misspellings, typos, ad-hoc abbreviations, and slang language. An active area of research is to normalize the language in social media in a way that is more traditional, by mapping out-of-vocabulary non-standard word to an in-vocabulary standard one and preserving the meaning of the original sentence [83, 84, 85].

2.2.2 Open-Vocabulary Approaches

Open-vocabulary approaches is a bottom-up approach, where signals derived from words are associated to outcome variables. Contrary to closed-vocabulary approaches, open-vocabulary approaches do not limit the analysis to a dictionary of terms instead they let the data discover what is important. As mentioned in Chapter 1, these include lexical features such as n-grams at the word and character level. The open-vocabulary approach is more suitable for tasks which may not necessarily be talk about as openly or frequently as the flu, such as human immunodeficiency virus (HIV) [86, 87]. The ease-of-use and portability of these features has made them popular in disciplines such as psychology, social science, journalism, and computer science [56, 88, 89, 90]. In [91], the authors use linguistic features such as words and phrases as well as semantic features, i.e., topic features to correlate user's Facebook posts with a volunteer collected personality measures. Features such as unigrams, bigrams, word occurrence counts, and location have been used to categorize anti-vaccine tweets [92]. An issue with open-vocabulary word count approaches, is that it is impossible fully capture all words with limited training data. Some tokenization approaches have been proposed to limit this factor, but as language evolves in social media text, there will always be new unseen words and concepts[22, 23, 24, 25]. A possible solution for this concept drift is to concurrently update the model once it becomes out of data [93], however this can become expensive and time consuming if the models are large.

2.2.3 Representation features

One alternative to open-vocabulary count-based methods is to learn better representations for the terms, such as new word representations based on the contexts the words are used in. A more recent line of work is representation learning, where features and feature representations are automatically learned in an optimization framework. Some examples include probabilistic latent semantic indexing [16], explicit semantic analysis [94] and latent Dirichlet allocation [32]. Most notably is word embeddings [95], where word semantics can be represented in a low dimensional real-valued vector space. Representation learning has applied beyond word-level embeddings, it is typically used to represent queries in question answering, documents [96] for classification.

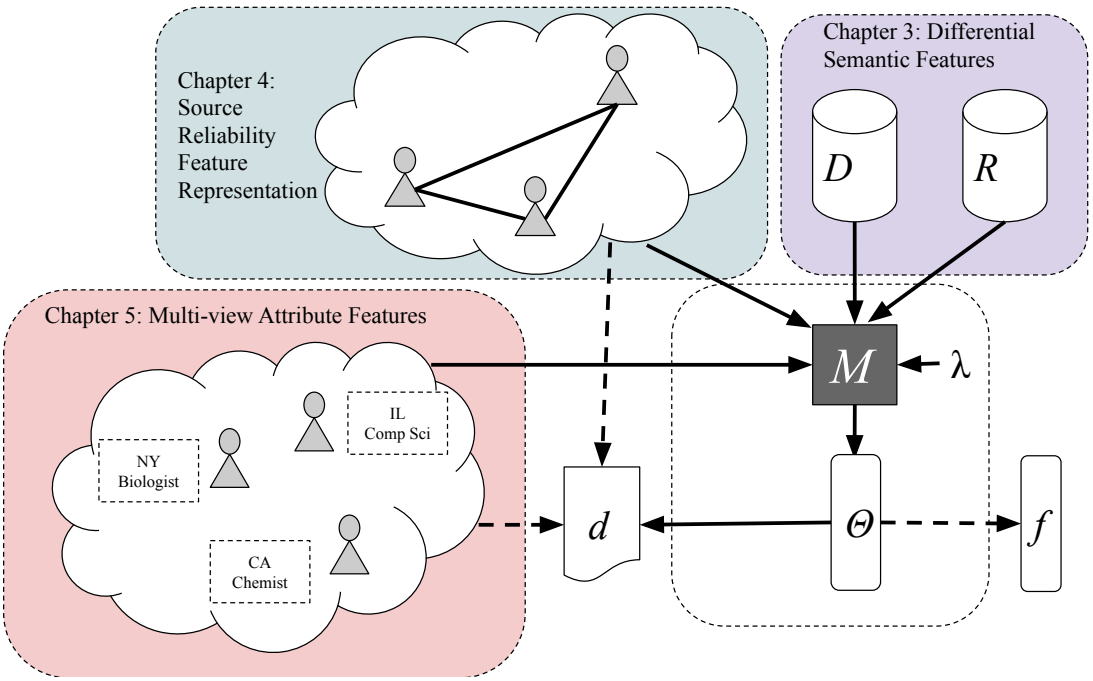


Figure 2.2: Model-based feature construction under different context of social media text. The context is color coded in the figure; in purple we have a reference corpus, in cyan we have user post network, and in red we have meta information of the user and the networks.

2.3 FEATURE CONSTRUCTION BEYOND TEXT DATA

While text could provide a valuable signal for prediction, it is rarely the case where text occurs alone and independently of any other forms of data. In Figure 2.2, we show the three themes for our work in this thesis in the model-based feature construction framework. In the figure, R is a reference corpus, which is used by the model to derive differential semantic features. The source reliability features use the user network information, such as commenting patterns to derive fine-grained reliability features. Finally, the multi-view features use associated information beyond the text data to tackle feature construction issues such as the text misalignment problem.

In this setting we categorize the approaches into two different approaches. In the first case, the methods can leverage additional companion text. For example, in this case methods might take advantage of external resources such as Wikipedia to augment the text data. This approach would add additional context to the original data since it could ground the text, also potentially unstructured, onto something that is more structured.

The second case involves using some available meta data, to further improve the type of text organization and thus feature construction. In social media text is just one facet of the signals produced by users. For example, in review data, such as Yelp, there is also some

additional user information such as review history, statistics about the user, restaurant as well as additional real-world location information.

CHAPTER 3: DIFFERENTIAL SEMANTIC FEATURE REPRESENTATION

In this chapter we describe a feature representation set derived from the comparison with a reference text source. The reference text source may be an external text source, or in-domain data. We call this *differential features*, and this type of feature has many different applications, for example identifying text topics, it is useful to compare with reference corpus of the same topics. Another example is author attribution, where the goal is to predict the author from a given some piece of text. Comparing these source text with some reference text can allow for abstraction of the shared information and thus focus on the distinguishing aspects of text.

In this chapter we describe differential semantic feature representation and apply it to humor identification task. In this task the documents have similar topic categories, that is they all review different aspects about a restaurant. However, by applying the differential features we can construct new representations based on incongruity, or unexpectedness of the text, allowing us to model humor. More specifically, we propose a generative language model, based on the theory of incongruity, to model humorous text, which allows us to leverage background text sources, such as Wikipedia entry descriptions, and enables construction of multiple features for identifying humorous reviews. Evaluation of these features using supervised learning for classifying reviews into humorous and non-humorous reviews shows that the features constructed based on the proposed generative model are much more effective than the major features proposed in the existing literature, allowing us to achieve almost 86% accuracy. These humorous review predictions can also supply good indicators for identifying helpful reviews.

3.1 DIFFERENTIAL FEATURE REPRESENTATIONS

A *differential feature* is a feature which capture the differences between a source text instance and some reference corpus. The work by Massung and Zhai in [97], SyntacticDIFF can create a text representation based on edit distances, such as insert, delete, and replace. This representation can then be used to distinguish native speaker and non-native speakers in text. Their work is an example of *differential syntactic feature* construction, as the representation relies solely on the edit distance to transform a source text to a reference corpus. Early work on summarizing the effects of code modifications has also leveraged syntactic differential feature construction [98]. In this domain the goal is to leverage previous version of the code and compare the syntax tree representations with the current version to

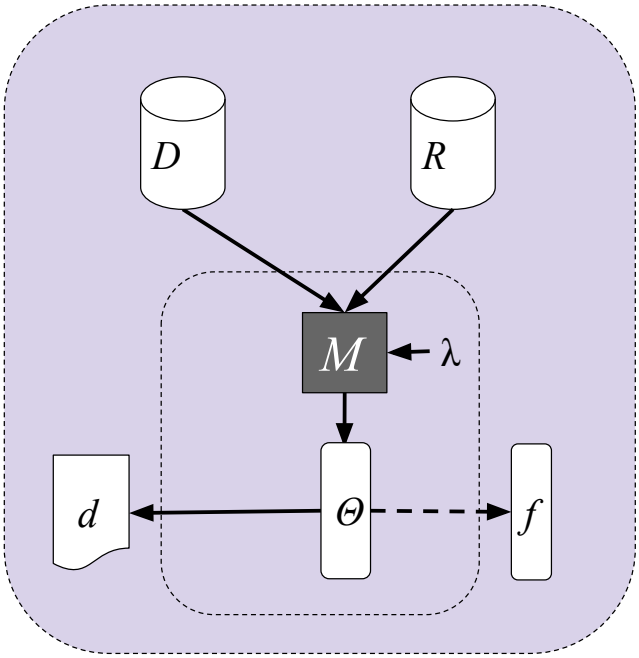


Figure 3.1: Model-based feature construction under reference corpus context of social media text. Differential features are derived from the comparison from a reference text corpus and a source text instance and this is represented by the two incoming arrows to the black box model $\mathcal{M}$ since the exact method to do this can vary.

identify and summarize the modification effects.

In contrast of differential syntactic features, *differential semantic features* are feature which capture the semantics and explainability of the source text by some reference text. In Figure 3.1, we show an abstract model-based feature construction representation for differential features in which the model depends on both the source text, but also a reference text corpus.

The earliest work proposing comparative text mining (CTM) analysis is by Zhai et al. in [99], which proposes a generative mixture model for both cross-collection and within collection clustering. The CTM model can discover common themes, as well as collection specific themes which can distinguish document clusters. However, the themes discovered are coarse, and also requires document clusters to discover these themes.

In [100], they propose multi-grained topic models to extract local and global topics, the local topics could be used as ratable aspects such as price, location, and decor for a restaurant review. [101] goes beyond this to predict the exact rating scores. These models model different aspects corresponding to the same corpus; however, it is often the case that we can leverage external or background text.

3.2 BACKGROUND TEXT FOR HUMOR IDENTIFICATION

An essential component for personal communication is the expression of humor. Although many people have studied the theory of humor, it remains loosely defined [102], this leads to difficulties in modelling humor. While the task for identifying humor in text has been previously studied, most approaches have focused on shorter text such as Twitter data [103, 104, 105] (see Section 3.6 for a more complete review of related work). In this chapter, we study the problem of automatically identifying humorous text from a new kind of text data, i.e., online reviews.

One possible formulation of humor identification in online reviews is to try to answer the question of "how funny is the review?". An issue with this formulation is that people's judgements are not always calibrated, it can also be the case that a review may have received more funny votes because it was more popular, or it was more visible.

In this chapter we try to explain what makes a piece of text, in our case online reviews, humorous. Specifically, we look at the classification problem of distinguishing funny/humorous reviews and non-funny/humorous reviews. The formulation of contrasting analysis may provide us a better opportunity to understand what makes a review humorous. In order to quantitatively test whether the review is humorous, we devise a novel approach, using the theory of incongruity, to model the reviewer's humorous intent when writing the review. The theory of incongruity states that we laugh because there is something incongruous [106], in other words, there is a change from our expectation.

Specifically, we propose a general generative language model to model the generation of humorous text. The proposed model is a mixture model with multinomial distributions as component models (i.e., models of topics), similar to Probabilistic Latent Semantic Analysis [16]. However, the main difference is that the component word distributions (i.e., component language models) are all assumed to be known in our model, and they are designed to model the two types of language used in a humorous text, including 1) the *general background model* estimated using all the reviews, and 2) the reference language models of all the *topical aspects* covered in the review that capture the typical words used when each of the covered aspects is discussed. Thus, the model only has the parameters indicating the relative coverage of these component language models. The idea here is to use these parameters to assess how well a review can be explained by collectively by the reference language models corresponding to all the topical aspects covered in the review, which are estimated using an external text source (e.g., Wikipedia). Thus, incongruity can be measured in two different ways, first if the text of a review that covers aspects has a relatively high likelihood of being generated by mixing all these reference language models (instead of being generated

by the background language model), it would be regarded as lacking incongruity. Whereas, if the reference language models cannot model the review well (i.e., the background language model must be used heavily to explain the review text), we could assume there exists incongruity since some vocabulary mentioned about an aspect must be inconsistent with the corresponding reference language model (causing the need to use more of the background language model). Secondly the usage of reference language models may provide a hint on what the intent for the review, if a review overly focuses on a single reference (i.e., the reference language model is heavily used to explain the review), then its highly incongruous. As opposed to if a review were to focus on the references equally, this means that the reviews mention references but focus on none of them.

We construct multiple features based on the generative model and evaluate them using supervised learning for classifying reviews into humorous and non-humorous reviews. Experiment results on a Yelp[1] review data set show that the features constructed based on the proposed generative model are much more effective than the major features proposed in the existing literature, allowing us to achieve almost 86% accuracy. We also experimented with using the results of humorous review prediction to further predict helpful reviews, and the results show that humorous review prediction can supply good indicators for identifying helpful reviews for consumers.

3.2.1 Graphical Modeling Review

The basis of our model is a graphical model, thus in this section we provide some background and introduction on probabilistic graphical modeling. A graph is a pair $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of vertices, and $\mathcal{E}$ is a set of paired vertices, $e = (v_i, v_j)$, where $v_i, v_j \in \mathcal{V}$ and e is an edge. $\mathcal{G}$ is a directed graph, if the edges have an orientation and we say v_i is a parent of v_j if there is an edge from with the respective start and end points. One key property of Bayesian networks is that the joint distribution of the variables in the graph is given by the product, over all the nodes in the graph, of a conditional distribution conditioned on the parents of the nodes [7]. We can write this factorization as follows

$$P(v) = \prod_{v_i \in \mathcal{V}} P(v_i | par(v_i)) \tag{3.1}$$

where $par(v_i)$ returns the parents of v_i.

In this introduction we use PLSA, a Bayesian network, as a running example as our

[1] www.yelp.com

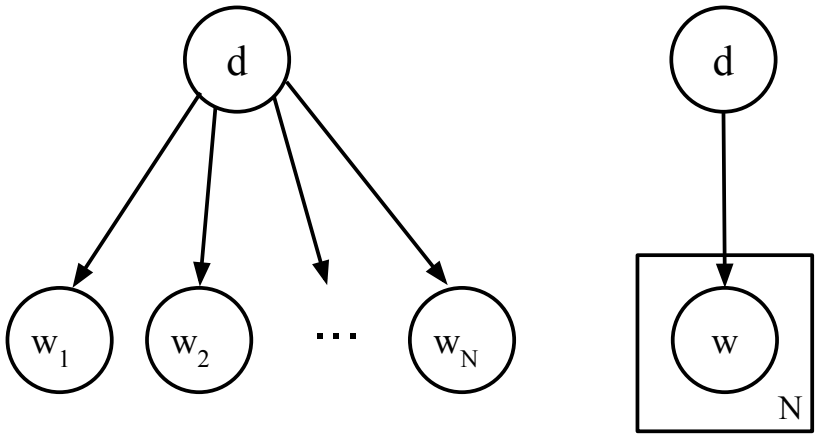


Figure 3.2: The left side represent a document with N words as a graphical model, and in the right side we represent the same model using plate notation to compress this representation.

proposed model is closely associated to this model. We use the *plate notation* convention of representing such a network, that is, instead of drawing a circle for every repeated variable i.e., word, we use a plate (or rectangle) to group those variables in a subgraph, with the corresponding number of repetitions inside the plate. In Figure 3.2, we show an equivalent representation of a document with N words under plate notation. In Figure 3.2 and Figure 3.3, d is a variable representing the document index, and w is a variable for the words in document d.

We represent PLSA as a plate notation in Figure 3.3, where z_w is the words topic drawn from $P(z|d)$. The shaded nodes indicated the observed variables, and the non-shaded nodes indicate the latent variables. We call a directed graph with no cycles, a directed acyclic graph, or DAG, note that graphs in Figure 3.2 and in Figure 3.3 are DAGs. Graphical models capture the casual process by which observed data is generated [107]. With the advent of big data, casual discover has been applied in many domains including genomics, ecology, epidemiology, biology, neuroscience, and social science [108].

As in graphical model representation of Bayesian networks, we can infer the variable's conditional independence directly from the graphical model through the d-separation property for directed graphs [107].

One way which we can simplify complex marginal distributions over observed variables is

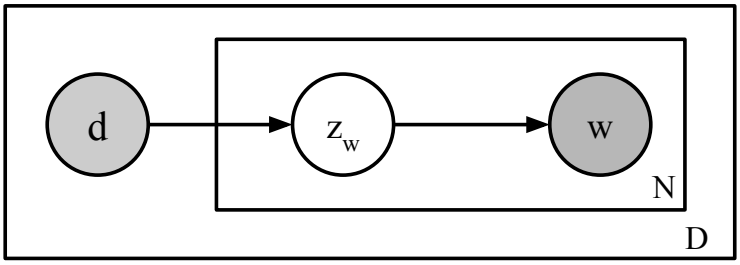


Figure 3.3: The plate notation associated with PLSA model, where d represents a document index, z_w corresponds to the topic drawn for word w.

to allow for the inclusion of hidden or latent variables. In doing so we may express these complex distributions in terms of joint distributions of both observed and latent variables. From Equation 3.1, we can write the factorization of Figure 3.3 as follows.

$$P(w, d) = P(d) \prod_{i=1}^{N} P(w_i|d) \tag{3.2}$$

PLSA models co-occurrence as a mixture of multinomial distributions, thus we can write the joint probability as follows,

$$P(w, d) = \sum_{z} P(w, z, d) = \sum_{z} P(d)P(z|d)P(w|z). \tag{3.3}$$

In order to estimate the parameters of the model, the EM algorithm can be used [109].

3.2.2 Referential Humor and Incongruity

In this section we describe some observations in our data that have motivated our approach to solving the problem. We show that humorous reviews tend to reference aspects which deviate from what is expected. That is, in funny reviews, the authors tend to use referential humor, in which specific concepts or entities are referenced to produce comedic effects, which we call *aspects*. Here we define *referential humor* to be a humorous piece of text which references aspects outside of the typical context, in our case restaurant reviews. For the rest of the dissertation, we use humorous and funny interchangeably.

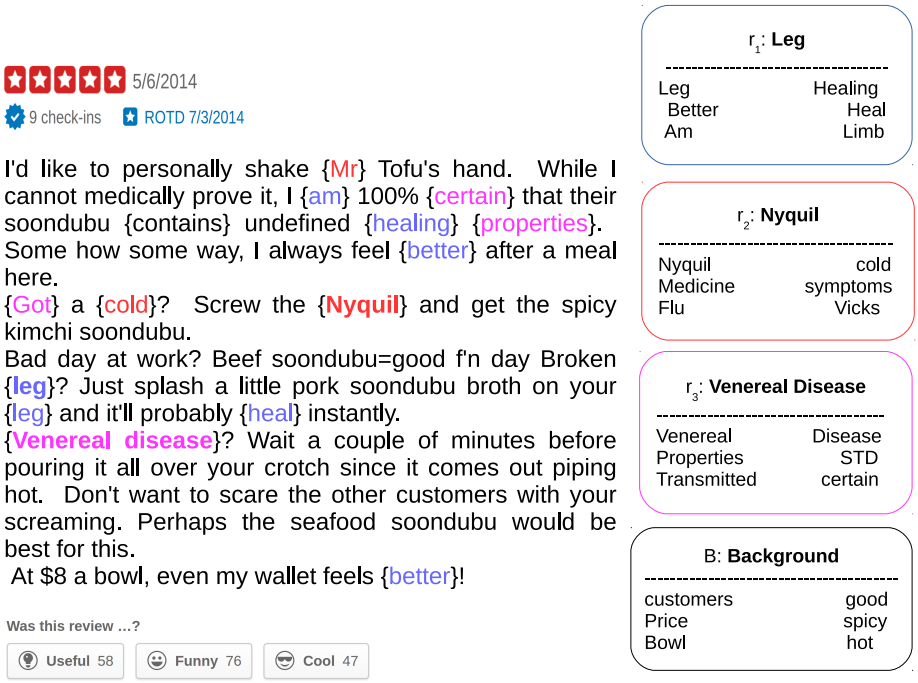


Figure 3.4: A funny review (left), with $K_d = 3$, aspect topics (right) contain words in their corresponding language model, probabilities removed for clarity, the colored (bracketed) word correspond to a different aspect assignment.

Our study uses review data from Yelp. Yelp has become a popular resource for identifying high quality restaurants. A Yelp user can submit reviews rating the overall experience of the restaurants. The reviews submitted to Yelp tend to have similar context, they mention several aspects rating the quality of the restaurant such as food, price, service and so on. This information is expected from the reviewer in their review; however, it is not always the case since there is no requirement for writing the review. Yelp users can vote for a review in several criterion, such as funny, cool, and useful. This gives the users an incentive for not only creating informative reviews but possibly entertaining reviews.

In Figure 3.4, we show a humorous review, randomly sampled by using our classifier with a high probability of being funny, where the reviewer asserts that the food has extreme medicinal properties. The reviewer refers to "Nyquil" a common cold medicine to express the food's incredible ability to cure ailments. This appears almost surprising since it would not normally be mentioned in restaurants reviews. Observing several reviews, we noticed that the humor in reviews often follows a similar structure. When reading a review about a restaurant, the reader is expecting to read about things such as the price, taste and quality of the food. To identify the intended humor, we can use the references the reviewer makes, e.g., Nyquil, as clues to what she is emphasizing, e.g., the savory soondubu, by making such

comparisons, e.g., the heavenly taste and amazing price.

> ... This dish was one to be savored.... no, to be fawned over and then savored. Using my meticulous chop-stick skills, I pampered each delicate noodle into my quivering mouth... which evoked blissful visions of lazily floating down the Chao Phraya river... and mind you, I've never even been to Thailand. ...

Figure 3.5: Humorous Review Excerpt

Yelp users seem to consider funny reviews which tended to deviate from what was expected into things which would seem out of place. For example, if we look at Figure 3.4, we see that typically when we read reviews, we wouldn't expect someone to talk about "Nyquil". Similarly looking at another excerpt in Figure 3.5 we notice the reviewer talking about the "Chao Phraya" river. These two items give us some intuition on why these reviews are considered funny.

3.3 REFERENCE LANGUAGE MODELS FOR INCONGRUITY FEATURES

Motivated by the observations discussed in the previous section (i.e., reviewers tend to reference some entities which seem unexpected in the context of the topic of the review), we propose a generative language model based on the theory of incongruity to model the generation of potentially humorous reviews. Following previous work on humor, we use the definition of incongruity in humor as "what people find unexpected" [103], where "unexpected" concepts are those concepts which people do not consider to be the norm in some domain, later we formalize unexpectedness using our model.

We now describe the proposed model in more detail. Suppose we observe the following references to K_d topical aspects $A_d = \{r_1, r_2, ..., r_{K_d}\}$ in a review $R_d = [w_1, w_2, ..., w_{N_d}]$, where each r_i corresponds to an aspect reference (i.e., NyQuil in our running example), and $w_i \in V$, where V is the vocabulary set. The model generates a word, for some review, at a time, which talks about a specific aspect or is related to the language used in Yelp more broadly; we call the latter the background language model. Thus, a word is generated from a mixture model, and its probability is an interpolation of the background language and the language of the references as shown in Figure 3.6. Conceivably, the background language model could be extended to capture many themes or sub-topics related to restaurant reviews.

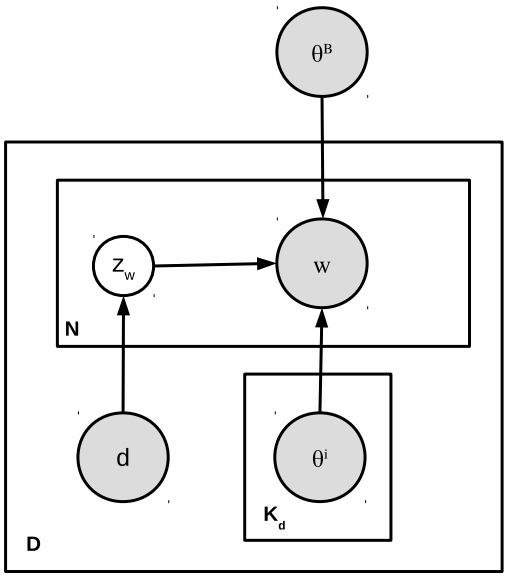


Figure 3.6: Generation model for text, where the dth document has K_d aspects in the text document. The shaded nodes here are the observed data and the light node z are the latent variables corresponding to aspect assignments.

These aspects provide some context to the underling meaning of a review; the reviewers use these aspects for creative writing when describing their dining experience. These aspects allow us to use external information as the context, thus we develop measures for incongruity addressing the juxtaposition of the aspect's context and the review. The review construction process is represented in a generative model, see Figure 3.6, where the shaded nodes represent our observations, we have observed the words as well as the referenced aspects which the reviewer has mentioned in their review. The light nodes are the labels for the aspect which has generated the corresponding word. Since the background language model, denoted by θ^B, is review independent, we can simplify the generative model by copying the background language model for each review, thus we can focus on the parameter estimation for each review in parallel.

A key component to the success of our features is the mesh of background text from external sources, or *background text sources*, and the reviews. In our example, Figure 3.4, Nyquil is a critical component for understanding the humor. However, it is difficult to understand some references a reviewer makes without any prior knowledge. To do so, we incorporate external background knowledge in the form of language models for the referenced aspect present in the reviews. If the reviewer has made K_d references to different aspects A_d in review R_d, then for each r_i there is a corresponding language model $\theta_w^{r_i} = P(w|\theta^{r_i})$ over the vocabulary $w \in V$. For simplicity, we describe the model for each document, and use the notation θ_w^i and θ^i for the corresponding language model of r_i.

3.3.1 Incorporating Background Text Sources

As described before, some features we will use to describe incongruity correspond to the weights of the mixture model used to generate the words in the review, which consider the language of the references she will make or allude, as shown in Figure 3.6. The probability that an author will generate a word w, for the dth review given corresponding aspects $\Theta = \{\theta^B, \theta^1, ..., \theta^{K_d}\}$, is

$$P(w, d, \Theta) = \sum_{z=0}^{K_d} P(w, z, d, \Theta) \tag{3.4}$$

$$= \sum_{z=0}^{K_d} P(w|z, \Theta)P(z|d) \tag{3.5}$$

$$= \lambda\theta_w^B + (1 - \lambda) \sum_{i=1}^{K_d} \pi_i\theta_w^i \tag{3.6}$$

Note K_d indicates the different aspects the reviewer will mention in a review, R_d, hence it can vary between reviews, and

$$\theta_w^B = P(w|z = 0, \Theta). \tag{3.7}$$

In Equation 3.6, θ_w^B is the probability that the word will appear when writing a review (e.g. background language model), and θ_w^i can be interpreted as word distributions over aspect i. Here

$$\lambda = P(z = 0|d) \tag{3.8}$$

is the weight for the background language model and

$$\pi_i = \frac{P(z = i|d)}{1 - P(z = 0|d)} \tag{3.9}$$

denotes the relative weights of the referenced aspect's language models used in the review. We denote our parameters for review R_d as

$$\Lambda_{R_d} = \{\pi_1, ..., \pi_{K_d}, \lambda\}. \tag{3.10}$$

Note that the parameter set varies depending on how many references the review makes. We show the computation graph of a review with three references in Figure 3.7. In this figure

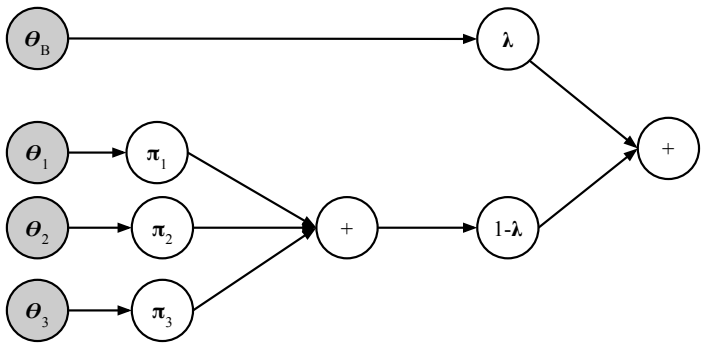


Figure 3.7: Example of parameters we estimate for a single review with 3 reference and the background text as a simplified computation graph. The arrows to the estimators indicate multiplication and the nodes with + indicate summation of all incoming variables.

we assume the corresponding term each language model is associated (the gray nodes) with is known and fixed. In order to estimate $P(w|\theta^i)$, we first need to find the aspects that the user is mentioning in their reviews. In general aspects can be defined as any topics explicitly defined in external background text data; in our experiments we define aspects as Wikipedia entities. In subsection 3.5.1, we describe one way of obtaining these aspects, but first we describe the estimation methodology.

3.3.2 Parameter Estimation

To estimate our parameters Λ_{R_d}, we would like to maximize the likelihood of $P(R_d)$, which is the same as maximizing the log-likelihood of $P(R_d)$. That is

$$\hat{\Lambda} = \text{argmax}_{\Lambda} \log P(R_d|\Lambda) \tag{3.11}$$

$$= \text{argmax}_{\Lambda} \sum_{w \in V} c(w, R_d) \log \left(P(w, d, \Theta) \right). \tag{3.12}$$

Here $c(w, R_d)$ represents the number of occurrences of the word w in R_d. In order to maximize the log-likelihood we use the EM algorithm [110], to compute the update rules for the parameters λ and $\pi_1, ...\pi_{K_d}$. For the E-Step, at the $n+1$th iteration we have

$$P(z_w = 0) = \frac{\theta_w^B \lambda^{(n)}}{\left(\sum_{l=1}^{K_d} \theta_w^l \pi_l^{(n)} \right) (1 - \lambda^{(n)}) + \theta_w^B \lambda^{(n)}} \tag{3.13}$$

$$P(z_w = j) = \frac{\theta_w^j \pi_j^{(n)}}{\sum_{l=1}^{K_d} \theta_w^l \pi_l^{(n)}} \tag{3.14}$$

Where z_w is a hidden variable indicating whether we have selected any of the aspect language models, or the background language model, when generating the word w. The update rules for the M-Step are as follows:

$$\lambda^{(n)} = \frac{\sum_{w \in V} c(w, R_d) P(z_w = 0)}{n}, \tag{3.15}$$

$$\pi_j^{(n)} = \frac{\sum_{w \in V} c(w, R_d) P(z_w = j)(1 - P(z_w = 0))}{\sum_{l=1}^{K_d} \sum_{w \in V} c(w, R_d) P(z_w = l)(1 - P(z_w = 0))} \tag{3.16}$$

We ran EM until the parameters converged or a small threshold was reached. Note there is some similarity to other topic modelling approaches like PLSA [16]. PLSA is a way to soft cluster the documents into several topics, in doing so a word distribution for each topic is learned. In our work we assume that the "topics" are fixed, namely they are the aspects which the reviewer mentions in their review. Note that, we can similarly derive update rules for a different topic model such as LDA [32], however prior work, [111], shows that LDA does not show superior performance over PLSA empirically for several tasks.

3.4 FEATURE CONSTRUCTION FOR HUMOR IDENTIFICATION

Since we are interested in studying discriminative features for humorous and non-humorous reviews, we set up a classification problem to classify a review into either humorous or non-humorous. In classification problems the data plays a critical role, for our task the labels are obtained from the funny votes in our Yelp dataset, and we describe how we created the ground-truth in Section 3.5. Here in this section, we discuss the new features we can construct based on the proposed language model and estimated parameter values.

3.4.1 Incongruity features

A natural feature in our incongruity model is the estimated background weight, λ, since it indicates how much emphasis the reviewer puts in their review to describe the referenced aspects, we denote this feature by **A1**. Another feature is based on the relative weights for the referenced aspect's language models. There tends to be more 'surprise' in a review when the reviewer talks about multiple aspects equally, this is because the more topics the reviewer writes about the more intricate the review becomes. We use the entropy of the

weights

$$H(R_d) = -\sum_{i=1}^{K_d} \pi_i \log \pi_i \tag{3.17}$$

as another incongruity score and label this feature as **A2**.

3.4.2 Unexpectedness features

Humor often relies on introducing concepts which seem out of place to produce a comedic effect. Thus, we want to measure this divergence from the references and the language expected in the reviews. Hence a natural measure is the KL-divergence measure the distance between the background language model and the aspect language models. We use the largest deviation,

$$\max_i \{D_{KL}(\theta^i || \theta^B)\} \tag{3.18}$$

as feature **D2**. For this feature we tried different combinations such as a weighted average, but both features seemed to perform equally so we only describe one of them.

By considering the context of the references in the reviews we can distinguish which statements should be considered as humorous, thus we also use the relative weight for each aspect to measure unexpectedness. Formally we have

$$U_j = \pi_j D_{KL}(\theta^j || \theta^B) \tag{3.19}$$

lastly we will denote $\max_i\{U_i\}$ these set of features as **U2**.

3.4.3 Baseline features from previous work

For completeness, we also include a description of all the baseline features used in our experiments; they represent the state of the art in defining features for this task. These features described below do not use any external text sources (leveraging external text sources is a novel aspect of our work), and they are more contextual and syntactical based features. We describe some of the most promising features, which have previously shown to be useful in identifying humor in text.

Context features: Due to the popular success of context features by [112] we tried the following features content related features:

- **C1**: the uni-grams in the review.[2]
- **C2**: length of the review.
- **C3**: average word length.
- **C4**: the ratio of uppercase and lowercase characters to other characters in the review text.

Alliteration: Inspired by the success that Mihalcea and Strapparava [103] had using the presence and absence of alliteration in jokes, we developed a similar feature for identifying funny reviews. We used CMU's pronunciation dictionary [3] to extract the pronunciation to identify alliteration chains, and rhyme chains in sentences. A chain is a consecutive set of words which have similar pronunciation, for example if the words "scenery" and "greenery" are consecutive they would form a rhyme chain. Similarly, "vini, vidi, visa" also forms another chain this time an alliteration chain. We used the review's total number of alliteration chains and rhyme chains and denote it by **E1**. Note that there could be different lengths of chains, we experimented with some variations, but they performed roughly the same, for simplicity we did not describe them here.

Ambiguity: Ambiguity in word interpretation has also been found to be useful in finding jokes. The reasoning is that if a word has multiple interpretation it is possible that the author intended another interpretation of the word instead of the more common one. We restricted the words in the reviews to only nouns and used Wordnet [4] to extract the synsets for these words. Then we counted the average number of synsets for each of these words, finally we took the mean score for all the words in the reviews. We call these features lexical ambiguity and denote it by **E2**.

3.5 HUMOR PREDICTION EXPERIMENTS

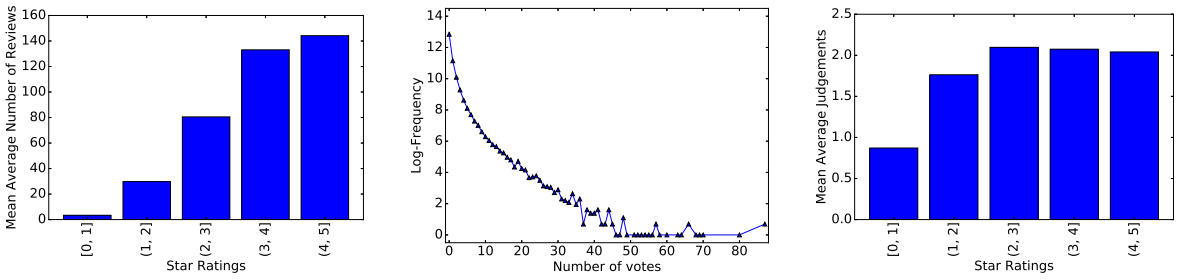
For our experiments we obtained the reviews from the Yelp Dataset Challenge[5], this dataset contains over 1.6 million reviews from 10 different cities. We also crawled reviews from Yelp in the Los Angeles area which is not included in the Yelp Dataset Challenge. This dataset was particularly interesting since the readers can vote whether a review is

[2]We also considered content-based features derived from PLSA topic weights, however the unigram features outperform these features.

[3]`www.speech.cs.cmu.edu/cgi-bin/cmudict`

[4]`http://wordnet.princeton.edu/`

[5]`http://www.yelp.com/dataset_challenge`



(a) (b) (c)

Figure 3.8: (a) Mean average number of reviews for restaurants falling in five different star rating ranges. (b) Log occurrences of funny votes per review. (c) Mean average voting judgements for restaurants in different star ratings.

considered *cool*, *funny*, and/or *helpful*. It also allows the flexibility for the reviewers to write longer pieces of text to express their overall rating of a restaurant.

3.5.1 Identifying Aspects in Reviews

We use recent advancements in Wikification, which aims to connect important entities and concepts in text to Wikipedia, it is also known as disambiguation to Wikipedia. We use the work of [113], in order to obtain the Wikipedia pages of the entities in the reviews, we call these *aspects* of the review. Using the Wikipedia description of the aspects we can compute the language models for each aspect. Using *mitlm*, the MIT language modeling toolkit by [114], we apply Modified Kneser-Ney smoothing to obtain the language models from the Wikipedia pages obtained from review's aspects.

3.5.2 Dataset Preliminaries and Groundtruth Construction

In Figure 3.8 we give an account of data statistics based on a random sample of 500,000 reviews, focusing on the funny voting judgements and the star rating distributions. In Figure 3.8a, we notice that on average the highly rated restaurants tend to have more reviews. Since users prefer to dine in a restaurant expecting to get a better overall experience, they create a feedback loop on the reviews for those highly rated restaurants. This "rich-get-richer" effect has been also been recently observed in other social networks [115] and a more detailed analysis is out of scope of this dissertation. We observe that most of the reviews receive a low number of funny votes in Figure 3.8b, with $\mu = 0.55$, where μ is the average funny rating. Computing the restaurant's average funny votes, then taking the mean by the star ratings for each category range, see Figure 3.8c, which seems to be consistently increasing

across the different star ratings. Note that this also includes the restaurants with zero funny votes, by excluding these we found that the ratings were more consistently stable on about 2.1 votes. Thus, regardless of restaurant rating, the funny reviews distribution is stable on average. Considering the prevalence of noise in the voting process, we also analyzed those reviews with more than one funny vote ($\mu = 3.90$), and with more than two votes ($\mu = 5.54$).

To construct our ground-truth data, we took all the reviews with at least five funny votes, which indicates the review was collectively funny, and considered those as humorous reviews, to remove noise we considered all the reviews with zero funny votes as non-humorous reviews. We obtained 17,769 humorous reviews and 856,202 non-humorous, from which we sampled 12,000 reviews from each category, and another 5,000 reviews was left for a development dataset, to obtain a corpus with 34,000 reviews total. The reviews on average had about 15 sentences and about 198 words. In total we collected 2,747 Wikipedia pages with an average of about 247 sentences per page. In our work we focused on identifying distinguishing features and relative improvement in a balanced dataset and while the true distribution may be skewed, we leave the unbalance distribution study for future work.

Finally, we use five-fold cross validation to evaluate all the methods. Due to the success of linear classifiers in text classification tasks we are interested in studying the Perceptron and Adaboost algorithms, we also use a Naive Bayes classifier which has been shown to perform relatively well in humor recognition tasks [103]. We use the Learning Based Java (LBJava) toolkit by [116] for the implementation of all the classifiers and use their recommended parameter settings. For the Averaged Perceptron implementation, we use a learning rate of 0.05 and thickness of 5. In Adaboost, we choose BinaryMIRA as our weak learner to do our boosting on. We also consider SparseWinnow and SparseConfidenceWeighted to be our weak learner as well, but the boosting performance for those two learners is marginal on the development set.[6] All experiments were run on an Intel Core i5-4200U CPU with 1.60GHz running Ubuntu.

3.5.3 Predicting Funny Reviews

We report the results of the features in Table 3.1. First, we can compare the accuracies of the individual features. For the content related features, we see that the best feature is **C1**, which is consistent to what others have found in humor recognition research [112]. The other content related features are based on some popular features for detecting useful reviews, however we notice that in the humor context it is not very effective. The performance of

[6]Since our main goal is to understand the effectiveness of various features, we did not further tune these parameters since they are presumably orthogonal to the question we study.

Features		Classifiers		
		Naive Bayes	Perceptron	AdaBoost
Content Related Features	**C1**	69.92 (0.545)	57.62 (1.084)	69.44 (0.485)
	C2	51.33 (1.250)	50.35 (0.763)	50.56 (1.155)
	C3	50.86 (0.812)	50.00 (0.012)	50.59 (1.122)
	C4	53.85 (0.486)	50.03 (0.172)	51.41 (1.205)
Alliteration	**E1**	50.81 (0.408)	50.11 (0.301)	50.28 (1.195)
Ambiguity	**E2**	51.53 (0.677)	50.39 (0.857)	51.78 (1.533)
Incongruity	**A1**	81.32 (0.974)	81.32 (0.974)	81.32 (0.974)
	A2	83.68 (0.623)	83.68 (0.623)	83.68 (0.623)
Divergence Features	**D2**	84.55 (0.550)	83.68 (0.627)	84.23 (0.561)
Unexpectedness	**U2**	83.68 (0.627)	83.68 (0.627)	83.68 (0.627)
Combination features	**A1 + C1**	74.24 (0.466)	79.45 (0.682)	80.19 (1.512)
	A1 + D2	84.55 (0.549)	83.68 (0.627)	84.35 (0.548)
	A2 + C1	73.00 (0.452)	79.45 (0.682)	82.59 (1.162)
	A2 + D2	84.55 (0.549)	84.00 (0.579)	84.41 (0.496)
	D2 + U2	84.55 (0.549)	84.00 (0.579)	84.40 (0.549)
	A1 + D2 + U2	84.55 (0.550)	84.02 (0.574)	84.33 (0.562)
	A2 + D2 + U2	84.55 (0.550)	83.89 (0.593)	84.35 (0.590)
	D2 + U2 + C1	78.28 (0.545)	79.63 (0.534)	83.18 (1.109)
	A1 + D2 + C1	77.87 (0.661)	79.63 (0.534)	82.49 (0.641)
	A2 + D2 + C1	78.87 (0.546)	82.68 (0.353)	85.61 (0.900)
	A1 + D2+U2+C1	78.62 (0.671)	79.63 (0.528)	85.77 (0.843)
	A2 + D2+U2+C1	78.87 (0.546)	81.60 (0.703)	85.60 (0.968)

Table 3.1: Classification accuracies, using 5-fold cross validation, the 95% confidence is given inside the parenthesis.

the contextual features could indicate that humor is not specific to a particular context and thus comparing different context between humorous and non-humorous text will not always work.

For the alliteration and ambiguity features which were reported to be very useful in short text, such as one-liners and on Twitter, are not as useful in detecting humours reviews. The reason is clear since when writing a funny review, the reviewer does not worry about the limitation of text and thus their humor does not rush to a punchline. Instead, the reviewer can write a longer more creative piece, adhering to less structure. The features based on incongruity and unexpectedness, do well in distinguishing the funny and non-funny reviews. For incongruity the best feature is **A2**, achieving about the same accuracy as unexpectedness features of about 83% accuracy.

The best feature was **D2** achieving an accuracy of around 84% accuracy. The features seem to be consistent over all our classifiers. This indicates that incorporating background text sources to identify humor in reviews is crucial, and our features we can indirectly capture some common knowledge, e.g., prior knowledge. It provides evidence that humor in online reviews can be better categorized as referential humor [102] rather than shorter jokes. The results also suggest that we can use these features to help predict the style of humorous text. Specifically **D2** is better than ambiguity **E2** and other features for one-liner jokes, exploring this would be an interesting venue for future work.

When we combine our features for the classification task and find that the best combination is the incongruity features with the divergence features. We do not report the results for features **E1, E2** and other context features, **C2, C3, C4**, since their performance when combined with other features did not add to the accuracy of the more discriminant feature. The divergence feature **D2** plays a big role in the accuracy performance. This is in line with our hypothesis that the more uncommon language used the more it is possible to be for a humorous purpose.

AdaBoost performed the best out of all three classifiers achieving about 86% accuracy, especially when more features were added, the classifier was able to use this information for improvement. While Naive Bayes and the Perceptron algorithm did not make such improvement achieving about 85% accuracy.

3.5.4 Ranking Funny Reviews

From the data we noticed that funny reviews tend to be voted highly useful, we noticed a correlation coefficient of 0.77. Although it would have been easy to use the useful votes as a feature to determine whether the review is funny/not funny, these scores are only available after people have been exposed to these reviews. To test how well the features worked when identifying helpful reviews, in a more realistic setting, we formulated a retrieval problem. Given a set of reviews, $\mathcal{D} = \{R_1, R_2, ..., R_m\}$ and relevant scores based on usefulness, $U = \{u_1, u_2, ..., u_m\}$, is it possible to develop a scoring function such that we rank the useful reviews higher? For this task we used the classification output of Naive Bayes, $P(\text{funny}|R_i)$ where i is the current example under consideration, for our scoring function and trained with the best performing features in the original dataset. We used a with-held dataset crawled from restaurants in Yelp in the Los Angeles area containing about 1,360 reviews with 260 reviews labelled as helpful and the other reviews labelled as not helpful. To obtain the ground truth we used the useful votes in Yelp similar to how we constructed the funny labels, using a threshold of 5 votes minimum to be considered helpful. This experiment

K	Precision @ K
1	1.00
10	0.50
25	0.48
50	0.44
100	0.45
200	0.54

Table 3.2: Precision of useful reviews.

reveals two things about our features for detecting humorous reviews. First, we see that the precision is around 50%, see Table 3.2, this is more than two times better than random guess which is about 19% and second that our features can be used to filter out some useful reviews.

3.6 HUMOR RELATED WORK

Although there has been much work in the theory of humor by many linguists, philosophers and mathematicians [117], the definition of humor is still a debated topic of research [106]. There have been many applications from computational humor research; for instance, creating embodied agents using humor, such as chat bots, which could allow for more engaging interactions and can impact many domains in education [118]. Existing work on computational humor research can typically be divided into humor recognition and humor generation.

In humor generation, some systems have successfully generated jokes and puns by exploiting some lexical structure in the pun/joke [119, 120, 121]. The HAHAcronym project was able to take user inputs and output humorous acronyms and it achieves comical effects by exploiting incongruity [122]. Work in automatic generation of humor is limited to some domains, usually only generating short funny texts.

One of the earliest works on humor recognition in text data is the work of Mihalcea and Strapparave [103], trying to identify *one-liners*, I.e., short sentences with a humorous effect. They frame the problems as a classification problem and develop surface features (alliteration, antonym, and adult slang) as well as context related features. They ultimately proposed that additional knowledge such as, irony, ambiguity, incongruity, and commonsense knowledge among other things would be beneficial in humor recognition, but they do not further pursue these avenues. Although they can distinguish between humorous and non-humorous one-liners, in longer texts, such as, reviews it is not so clear that these features

suffice. More recent research uses deep learning for humor recognition [123]. Instead, we make use of the creative writing structure of the reviewers by leveraging the referenced entities in their reviews

Although verbal irony can be humorous, and an active topic of research [124], it is often defined as the "opposite to what the speaker means" and combining features for identifying both humor and irony has been studied (see, e.g., [104]). In the work by [104], the authors defined the unexpectedness feature as semantic relatedness of concepts in Wordnet and the assumption was that the less the semantic relatedness of concepts the funnier the text. In our work we use a similar definition but applying it to the "topical" relatedness of the referenced aspects and the background language model. The authors demonstrate that irony and humor share some similar characteristics and thus we can potentially use similar features to discriminate them. There has been some early work on identifying humor features in web comments [105], in these comments the users can create humor through dialogue thus making the problem more complex. More recently there was a workshop in SemEval-2017 [7], which focus is on identifying humorous tweets which are related, typically as a punchline, to a particular hashtag. SemEval-2020 [8], assess the funniness of edited news headlines, and they propose a more fine-grained scale of humor that includes "Not Funny", "Slightly Funny", "Moderately Funny" and "Funny" [125].

[126] aimed to understand "That's what she said" (TWSS) jokes, which they classify as double entendres. They frame the problem as metaphor identification and notice that the source nouns are euphemisms for sexually explicit nouns. They also make use of the common structure of the TWSS jokes to the erotic domains to improve 12% in precision over word-based features. In our work we try to explicitly model the incongruity of the reviewer, by doing so we are able to distinguish the separate language used by the user when introducing humorous concepts. Recently there has been work in consumer research, to identify the prevalence of humor in social media [127]. The focus was to examine the benign violation theory, which "suggest that things are humorous when people perceive something as wrong yet okay". One of their finding suggests that humor is more prevalent in complaints than in praise, thus motivating the usage of automatic humor identification methods for restaurants regardless of its popularity.

As described above, much research has focused on humor for short text and thus, there has also been a need for constructing larger datasets for humor [128]. In our work we construct a novel humor dataset which is comprised of longer pieces of texts that allows for creative writing in humor. While there is a breadth of work in identifying helpful reviews and opinion

[7] http://alt.qcri.org/semeval2017/task6/

[8] https://competitions.codalab.org/competitions/20970

spam in reviews [129] as well as deceptive opinion spam [57], and synthetic opinion spam [130]; we show that humor can also be used to identify helpful reviews.

3.7 SUMMARY

In this chapter we have introduced differential semantic features in the context of humorous text identification. We introduced a probabilistic generative model, which compares source text with background text. Our model introduces a novel and way to incorporate external text sources for humor identification task, and which can be applied to any natural language provided there is a reference database, i.e., news articles or Wikipedia pages, in that language. This model is then used to develop features which differentiate the source context, i.e., restaurant entities and referenced aspects.

In using a reference corpus, we can leverage sources beyond the domain text, i.e., $\mathcal{D}$ is different than our domain text E. Though this is applied in a social media setting, this only limited to sources which we can map to a reference corpus. As we will delve more in the next chapter, social media data is rich in its interconnections, that is text rarely occurs in isolation.

CHAPTER 4: SOURCE RELIABILITY FEATURE REPRESENTATION

A theme of this dissertation is representing text along with its context, in the previous chapter we showed that using text and background text sources can incorporate some common background knowledge to develop differential semantic feature representations. In this chapter we show that we can use text to mine latent aspects of sources. We develop a model for the problem of comment trustworthy identification which can also be used to develop *source reliability feature representations*.

4.1 SOURCE FEATURE REPRESENTATION

The source of text typically corresponds to an author. In social media, there are many reasons to construct *source feature representations*, for example in the case of recommender systems, leveraging the text which an author has written gives us clues about the things they are interested. In personalized recommendation systems, it is content-based models model the latent aspect about the user and products jointly. In [131], the authors develop a model which captures hidden factors and hidden topics for latent rating dimensions. The limitation for these methods is that they rely on the explicit rating, which may not necessarily be available to represent sources. One setting which is useful for source representation is in discussion forums, where we only observe the source posting patterns. The authors in [132], overcome this sparsity limitation on the common items reviews by users, and propose a user-preference-based collaborative filtering approach for personalized recommendation. They model both the aspect importance and aspect need for each user to measure the similarities and differences among the users. In the medical domain, researchers have applied a similar approach to recommend health services to users that take topical preferences as well as emotional offsets of users [133]. Modeling the user's preferences or expertise is useful for recommender systems, however it makes an important assumption that the reviews or text is reliable and equally credible which is not necessarily true.

4.1.1 Source Reliability Feature Representation

A special case of source feature representations is, *source reliability feature representations*, which can capture the reliability representations of each source. This is used in truth-discovery analysis, where the task is to identify truthful claims and reliable sources [134], or reliable users in community questioning answering forums [135, 136]. The notion of

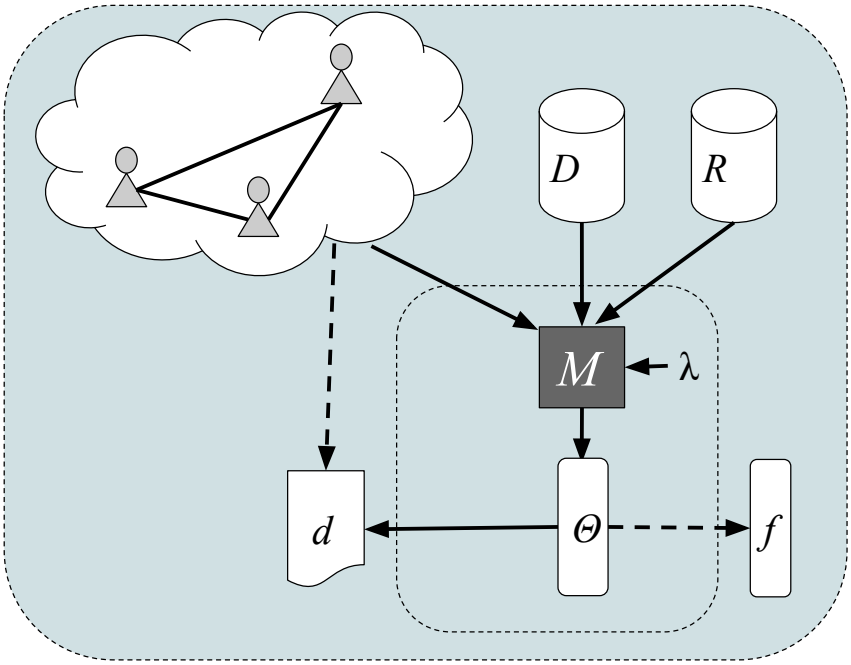


Figure 4.1: The source context modeling in model-based feature construction represented by a cloud.

reliability extends to misinformation detection where the goal is to identify truthful vs fake news, both in news articles and social media posts [137, 138]. Reliability can be viewed from a model-based feature construction method taking the context of sources. In Figure 4.1, we show the source context as a user-network cloud, in order to emphasize the social network aspect of this context. The source context modeling can also include reference text corpus in the model via background estimation, such as priors associated to the users from other text sources. Note that the user-network can be either explicit or implicitly derived from social media. For example, an explicit network may be a friend-network [139] or follow-networks [140]. In contrast, an implicitly derived network, could be derived from retweets on twitter [141, 142, 143, 144], by sharing content in social media [145], Question-Answering networks [146] and from comment-post patterns on discussion forums [54].

4.2 SOURCE RELIABILITY FEATURE REPRESENTATION FOR IDENTIFYING TRUSTWORTHY COMMENTS FROM COMMUNITY DISCUSSION FORUMS

As more and more people turn to online discussion forums to seek useful advice, the need for assessing the trustworthiness of user-generated responses has become imperative. Truth discovery methods estimate user reliability and infer information trustworthiness si-

multaneously, and thus can be used to identify trustworthy comment in an unsupervised manner. However, discussion forums typically encompass various topics and exhibit the diversity of expression in the users' comments, which the existing truth discovery methods do not capture. We thus propose our CrowdQM model that simultaneously estimates aspect-based reliabilities of users and semantic representations of words to learn embedding of the most trustworthy comment for each question. We verified our model to identify trustworthy comments for three diverse communities in Reddit. We show qualitative results on learned reliability scores and word embeddings of our model.

Users are increasingly turning to community discussion forums to solicit domain expertise, such as requesting for help about inscrutable political events on history forums or posting a health-related issue to seek medical suggestions or diagnosis. These forums provide users prompt feedback, instead of the possibly laborious and expensive alternatives such as researching a particular event or consulting a medical professional. Social media networks and online discussion forums have been shown to provide valuable insights; some examples include discovering opposing discussion opinions [45], user metrics, and community question answering [147].

More often than not, they contain subjective replies as well as personal or anecdotal experiences. While these forums may be useful, due to almost no regulations on post requirements or user background, most responses contain conflicting and unreliable information [53]. This misinformation could lead to severe consequences, especially in health-related forums, that outweighs the positive benefits of these communities. To address this challenge, some forums employ moderators to curate appropriate responses; however, it is not only expensive to curate each reply manually, but also unsustainable. Most of these discussion forums employ voting mechanisms to help users to infer the trustworthiness of the responses. However, there is widespread under-provision of votes, and thus, it is possible to miss high-quality content that is not highly voted [148]. In this chapter, we address this challenge by exploiting the truth discovery principle to simultaneously identify trustworthy comments and user reliability in an unsupervised manner while incorporating semantic similarity between comments.

Broadly, the general truth discovery principle is as follows: the more trustworthy information the user provides, the higher the reliability; more reliable users provide the same claim, more trustworthy is the information [149, 150, 151, 152, 153]. This principle underlines the importance to estimate trustworthy information and source reliability concurrently in unsupervised settings. However, these methods typically represent user reliability as a single real value, not considering the context/topic of the post [154]. While a single source-reliability score can distinguish users broadly, it may not be suited for forums which encompass diverse

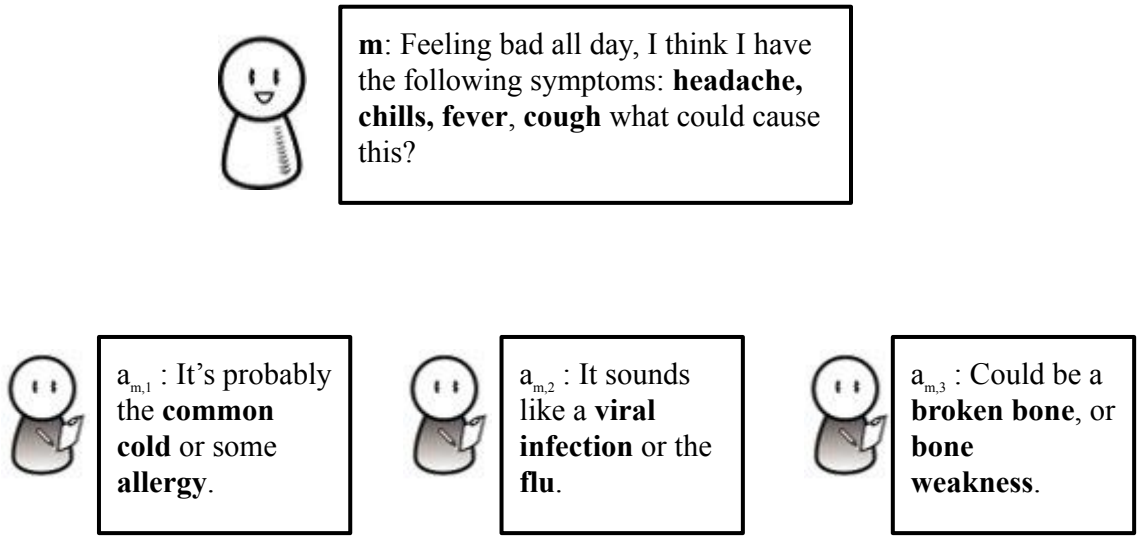


Figure 4.2: A toy example of a submission post and three comments for that post.

topics. This heterogeneity is especially true for discussion forums, like Reddit, which have communities catering to broad themes, where questions span a diverse range of sub-topics. For instance, in a science forum, a biologist could be highly knowledgeable, and in turn reliable, when she answers biology-related questions but may not be competent enough for linguistic queries. Motivated by this observation, we propose *aspect-based user reliability* model that allows us to learn reliability over fine-grained topics effectively.

Another challenge is the diversity of word expressions in the responses. Truth discovery-based approaches treat each response as categorical data. However, in discussion forums, users' text responses can include contextually correlated comments [155]. For instance, in the *context* of a post describing symptoms like "headache" and "fever", either of the related responses of a viral fever or an allergic reaction can be a correct diagnosis, see Figure 4.2. However, unrelated comments in the post should be unreliable; for instance, a comment giving a diagnosis of "bone fracture" for the above symptoms.

In discussion forums, users' text responses can include semantically correlated comments [155]. To account for this diversity of word expression, we capture semantic meaning of comments and post through word embeddings in our model. We then learn trustworthy comment embeddings such that it is similar to comment embeddings of reliable users and also to the post's context. Moreover, we update these word embeddings in a trust aware manner such that terms used by reliable users are closer in the embedding space.

In this work, we propose CrowdQM model that uses user aspect-based reliability and context similarity to identify most reliable responses for community discussion forums.

CrowdQM addresses both limitations by jointly modeling the aspect-level user reliability and latent trustworthy comment in an optimization framework while incorporating semantic similarity between words. By leveraging the mutual reinforcement heuristic discussed earlier, the framework can estimate aspect-level reliability scores for all users and produce embedding representation of the estimated trustworthy content simultaneously in an unsupervised way. Our model builds upon the truth discovery principle widely used to estimate information reliability in the presence of noisy information sources (users in our case). Compared with previous work, our framework has two novel features, both beneficial for many applications: In particular,

- CrowdQM learns user reliability over fine-grained topics discussed in the forum. This improved model of reliability can be further used to improve other user related tasks like expert finding which depend on the similarity with respect to the trustworthy comments and the context similarity of the answers, weighted by the appropriateness of the response, leading to a more detailed model of user expertise, which not only provides a more accurate model of user reliability and comment trustworthiness, but also enables many interesting ways to analyze user reliability.

- Our model captures the semantic meaning of comments and posts through word embeddings. We update these word embeddings in a trust aware manner, such that, terms used by only reliable users in similar post's context are closer in the embedding space.

We learn a trustworthy comment embedding for each post, such that it is semantically similar to comments of reliable users on the post and similar to the post's context. Contrary to the earlier approaches [156, 157, 158], we propose an *unsupervised model* for comment trustworthiness that does not need labeled training data.

We verified our proposed model on the trustworthy comment ranking task for three Ask* *subreddit communities*. Our model outperforms state-of-the-art baselines in identifying the most trustworthy responses, deemed by community experts and community consensus. We also show the effectiveness of our aspect-based user reliability estimation and word embeddings qualitatively. Furthermore, our improved model of reliability enables us to identify reliable users per topic discussed in the community.

4.3 CROWDQM

A challenge in applying truth discovery to discussion forums is capturing the variation in user's reliability and the diversity of word usage in the answers. To address it, we model

aspect level user reliability and use semantic representations for the comments.

4.3.1 Trustworthiness Comment Identification Problem Formulation

Each *submission* is a post, i.e., question, which starts a discussion thread while a *comment* is a response to a submission post. Formally, each submission post, m, is associated with a set of terms, c_m. A user, n, may reply with a comment on submission m, with a set of terms $w_{m,n}$. $\mathcal{V}$ is the vocabulary set comprising of all terms present in our dataset, i.e., all submissions and comments. Each term, $\omega \in \mathcal{V}$ has a corresponding word-vector representation, or word embedding, $v_\omega \in \mathbb{R}^D$. Thus, we can represent a post in terms of its constituent terms, $\{v_c\}, \forall c \in c_m$. To capture the semantic meaning, we represent each comment as the mean word-vector representation of their constituent terms[1]. Formally, we represent the comment given on the post m by user n as the *comment embeddings*,

$$a_{m,n} = |w_{m,n}|^{-1} \sum_{\omega \in w_{m,n}} v_\omega. \tag{4.1}$$

Our model treats the post word embeddings as static and learns the comment word embeddings. The set of posts user n has commented on is denoted by $\mathcal{M}_n$ and the set of users who have posted on submission m is denoted as $\mathcal{N}_m$.

There are total K aspects or topics discussed in the forum and each post and comment can be composed of multiple *aspects*. We denote submission m's distribution over these aspects as the *post-aspect distribution*, $p_m \in \mathbb{R}^K$. Similarly, we also compute, *user-aspect distribution*, $u_n \in \mathbb{R}^K$, learned over all comments of the user n in the forum. This distribution captures familiarity (or frequency) of user n with each aspect based on their activity in the forum. Each user n also has an *user reliability* vector defined over K aspects, $r_n \in \mathbb{R}^K$. The reliability captures the likelihood of the user providing a trustworthy comment about a certain aspect. Note high familiarity in an aspect does not always imply high reliability in the same aspect.

For each submission post m associated with a set of responses $\{a_{m,n}\}$, our goal is to estimate the real-valued vector representations, or *latent trustworthy comment* embeddings, $a^*_m \in \mathbb{R}^D$. We also simultaneously infer the *user reliability* vector $\{r_n\}$ and update the word embeddings $\{v_\omega\}$. The learned trustworthy comment embeddings, a^*_m, can then be used to rank current comments on the post. We summarize the symbols used in Table 4.1.

In Figure 4.3, we show the post-aspect distribution as a histogram plot, similarly we show

[1]Sentence, and furthermore document representation is a complex problem. In our work, we explore a simple aggregation method for comment semantic composition [159].

Notation	Definition
m	index for a submission post/question
n	index for a user/commenter
$w_{m,n}$	text comment from user n on submission m
$a_{m,n}$	embedding of comment from user n to post m
D	word vectors the embedding dimension
M	total number of posts
N	total number of users
K	total number of aspects
$\mathcal{V}$	vocabulary associated with submission/comments
$\mathcal{M}_n$	submissions where user n has commented
$\mathcal{N}_m$	users who have commented on submission m
$\mathcal{D}_\omega$	comment-submission pairs where ω term appears
p_m	post-aspect distribution of submission m
u_n	user-aspect distribution of user n
r_n	user reliability vector for user n
a^*_m	latent trustworthy comment embedding for post m
v_ω	word embedding for term ω
$a^{-\omega}_{m,n}$	embedding of comment from user n on post m, excluding term ω
$< m, n >$	comment-post pair
$r^{(k)}_n$	learned user-aspect reliability for user n for aspect k
$u^{(k)}_n$	kth user-aspect weight for user n
$p^{(k)}_m$	kth post-aspect weight for submission post m
β	question context weight parameter
$R_{m,n}$	the user-post reliability for the nth user and the mth post
$E_{m,n}$	the embedding error of the nth user's comment on the mth trustworthy comment representation
$Q_{m,n}$	the context error the nth user's comment and the mth post context

Table 4.1: Symbols used and their meaning.

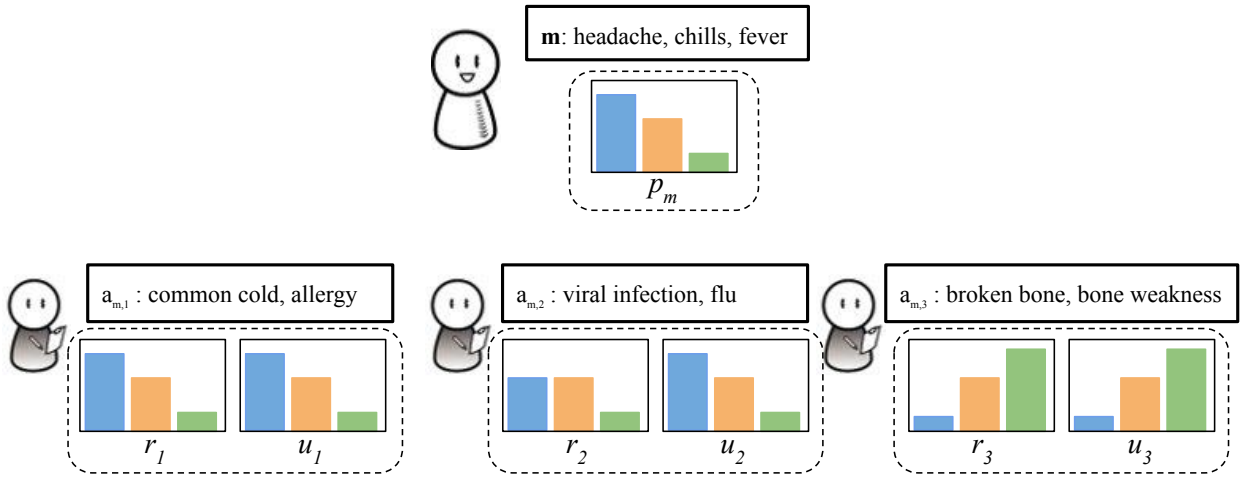


Figure 4.3: An extension of Figure 4.2 with the aspect distributions and user-aspect reliability distributions. The colors represent the same aspect across the submission post and comments, as well as the user reliabilities.

the user-aspect distributions and the corresponding user-aspect reliabilities. In this example the response $a_{m,1}$ is deemed both more similar to the post in terms of the aspect distributions, while the response $a_{m,3}$ is the least similar since it is about broken bones rather than flu-like diagnosis. The distinction between the first and second users is the user reliability associated to the first aspect, i.e., $r_1^{(1)} > r_2^{(1)}$, we next describe how we take this into account when finding the trustworthy comments.

4.3.2 Proposed Method

Our model is based on the following principles, the trustworthy comment should be semantically similar to the comments given for the post. To capture this, we need to minimize the *embedding error*,

$$E_{m,n} = ||a_m^* - a_{m,n}||^2, \tag{4.2}$$

i.e., mean squared error between learned *trustworthy comment* embeddings, a_m^* and comment embeddings, $a_{m,n}$, on the post m. This error ensures that the trustworthy comment is estimated from all diverse comments presented for post m.

Next, the comments, in turn, should be relevant to the context of the post. This is

computed by the *context error*,

$$Q_{m,n} = |c_m|^{-1} \sum_{c \in c_m} ||a_{m,n} - v_c||^2, \tag{4.3}$$

reducing the difference between the *comment embeddings* and *post embeddings*. The key idea is similar to that of the distributional hypothesis that if two comments co-occur a lot in similar posts, they should be closer in the embedding space.

Furthermore, these errors should depend on the reliability of the user providing the comment. We estimate the reliability of user n for the specific post m through the *user-post reliability* score,

$$R_{m,n} = r_n \odot s(u_n, p_m) = \sum_k r_n^{(k)} \cdot (u_n^{(k)} \cdot p_m^{(k)}). \tag{4.4}$$

The $\odot$ symbol represents the Hadamard product. This score computes the magnitude of *user reliability* vector, r_n, weighted by the similarity function $s(.)$. The similarity function $s(u_n, p_m)$ captures user familiarity with post's context by computing the similarity between the aspect distribution of the user n and the post m. We use the product operator as $s(.)$ in our experiments.[2] Thus, to get a high *user-post reliability* score, the user should both be reliable and familiar to the aspects discussed in the post.

Finally, these errors should be aggregated over all user's comments. Motivated by the above principles, we minimize the following objective function,

$$\min_{\{a_m^*\},\{v_\omega\},\{r_n\}} \sum_{n=1}^{N} \sum_{m \in \mathcal{M}_n} \underbrace{R_{m,n}}_{\text{user-post reliability}} \left(\underbrace{E_{m,n}}_{\text{embedding error}} + \beta \odot \underbrace{Q_{m,n}}_{\text{context error}} \right) \quad \text{s.t.} \sum_{n=1}^{N} e^{-r_n^{(k)}} = 1; \forall k \tag{4.5}$$

where N is the number of users. Thus, $R_{m,n} \cdot E_{m,n}$ ensures that the learned trustworthy comment embeddings are most similar to comment embeddings of *reliable* users for post m. While $R_{m,n} \cdot Q_{m,n}$ ensures trust aware learning of contextualized comment embeddings. The hyperparameter β controls the importance of context error in our method. The exponential regularization constraint, $\sum_{n=1}^{N} e^{-r_n^{(k)}} = 1$ for each k, ensures that the reliability across users are nonzero. Figure 4.4 shows the overview of our model using a toy example of a post in a medical forum with flu-like symptoms. The commenters describing flu-related diagnosis are

[2]Other metrics like cosine and product complement performed slightly worse.

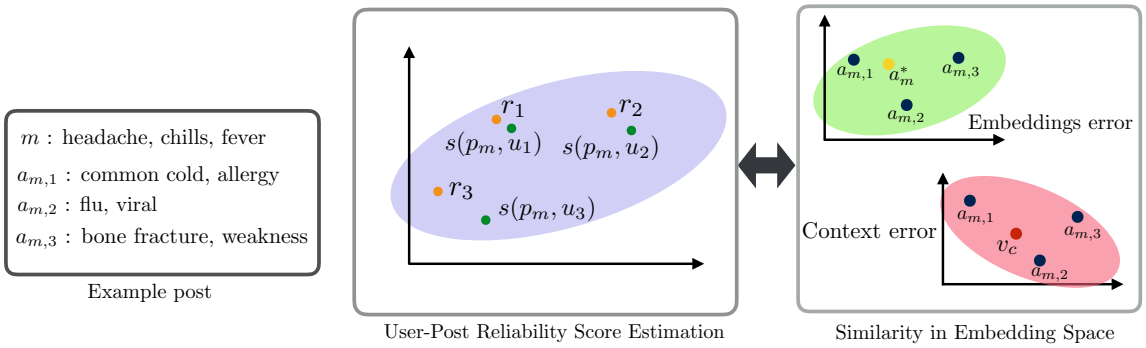


Figure 4.4: An illustrative toy example detailing our model components. The left-hand side details the user-post reliability estimation, $R_{m,n}$, that is a function of similarity function $s(.)$ between the user and post aspect distributions and user aspect reliabilities r_n. In the right-hand, we learn trustworthy comment embedding a_m^* such that they are similar to user comments, $a_{m,n}$ which are in turn similar to the post context v_c. Representative words are shown for question and answer for illustrative purposes. The most trustworthy comment representation is given by a_m^1, the aspect distribution of post, q_m, and comment, p_1 are alike; also, user-aspect reliability r_1 is high for those aspects.

deemed more reliable for this post.

4.3.3 Solving the Optimization Problem

We use coordinate descent [160] to solve our optimization problem. In particular, we solve the equation for each variable while keeping the rest fixed.

Case 4.1. Fixing $\{r_n\}$ and $\{v_\omega\}$, we have the following update equation for $\{a_m^*\}$:

$$a_m^* = \frac{\sum_{n \in \mathcal{N}_m} R_{m,n} a_{m,n}}{\sum_{n \in \mathcal{N}_m} R_{m,n}} \tag{4.6}$$

Thus, the learned *trustworthy comment* is a weighted combination of comments where weights are provided by the *user-post reliability* score $R_{m,n}$. Alternatively, it can also be interpreted as a reliable summarization of all the comments.

Case 4.2. Fixing $\{a_m^*\}$, $\{v_\omega\}$, we have the following update equation for $\{r_n^{(k)}\}$:

$$r_n^{(k)} \propto -\ln \sum_{m \in \mathcal{M}_n} s(u_n^{(k)}, p_m^{(k)}) \left(E_{m,n} + \beta Q_{m,n}\right) \tag{4.7}$$

Reliability of a user in aspect k is inversely proportional to the errors with respect to the learned trustworthy comment a_m^* ($E_{m,n}$) and submission's context v_c ($Q_{m,n}$) over all her posted comments ($\mathcal{M}_n$). The embedding error ensures that if there is a large difference

between the user's comment and the trustworthy comment, her reliability becomes lower. The context error ensures that non-relevant comments to the post's context are penalized heavily. In other words, a reliable user should give trustworthy and contextualized responses to posts.

This error is further weighed by the similarity score, $s(.)$, capturing familiarity of user with the post's context. Thus, familiar users are penalized higher for their mistakes as compared to the unfamiliar users.

Case 4.3. Fixing $\{a^*_m\}, \{r^{(k)}_n\}$, we have the following update equation for $\{v_\omega\}$:

$$v_\omega = \frac{\sum_{<m,n>\in D_\omega} R_{m,n} \left(a^*_m + \beta |c_m|^{-1} \sum_{c\in c_m} v_c\right) - R_{m,n}(\beta+1)|c_m|^{-1} a^{-\omega}_{m,n}}{\sum_{<m,n>\in D_\omega} R_{m,n}(\beta+1)} \tag{4.8}$$

where,

$$< m, n >\in D_\omega = \{(m,n) | \omega \in w_{m,n}\} \tag{4.9}$$

and

$$a^{-\omega}_{m,n} = |w_{m,n}|^{-1} \sum_{\omega' \in w_{m,n}\setminus\{\omega\}} v_{\omega'}. \tag{4.10}$$

To update v_ω, we only consider those comment and submission pairs, D_ω, in which the particular word appears. The update of the embeddings depends on the submission context v_c, learned trustworthy comment embedding, a^*_m as well as *user-post reliability* score, $R_{m,n}$. Thus, word embeddings are updated in a trust-aware manner such that reliable user's comments weigh more than those of unreliable user as they can contain noisy text. Note that there is also some negative dependency on the contribution of other terms in the comments. The full derivation of the updates can be found in Appendix A.1.

4.3.4 Implementation Details:

We used popular Latent Dirichlet Allocation (LDA) [32] to estimate aspects of the posts in our dataset[3]. Specifically, we combined title and body to represent each post. We applied topic model inference to all comments of user n to compute its combined aspect distribution, u_n. To compute the aspect distribution for each post, we treated its title and body as a single document and learned a topic model for these posts. We used Latent Dirichlet Allocation

[3]We ran LDA with 50 topics for all experiments and examined its sensitivity in Section 4.5.1.

Figure 4.5: Snapshot of the AskDocs subreddit submission post threads as of November 2020.

(LDA) [32] to derive these topical distributions. To generate the user-aspect distribution we applied topic model inference to user n's comments. We combined all the comments (over all the posts) made by user n into a single document and initialized the user weights, r_n by sampling from a random uniform distribution. We randomly initialized the user reliability, r_n. We initialized the word embeddings, v_ω, via word2vec [161] trained on our dataset. We used both unigrams and bigrams in our model. We fixed β to 0.15 we did not find significant change in results for different values of β The model converges after only about six iterations indicating quick approximation. In general, the computational complexity is $O(|\mathcal{V}|NM)$; however, we leveraged the data sparsity in the comment-word usage and user-posts for efficient implementation.

4.4 THE CROWDQM ASKREDDIT DATASET

We evaluate our model on widely popular discussion forum Reddit. Reddit has grown to be one of the most visited online social discussion site on the internet [4], with more than 330 million active users and more than half a million communities called subreddits. Reddit covers diverse topics of discussion and is challenging due to the prevalence of noisy responses.

[4]https://www.alexa.com/topsites/countries/US

We specifically tested on *Ask** subreddits as they are primarily used to seek answers to a variety of topics from mundane issues to serious medical concerns.[5] In Figure 4.5 we show a snapshot of the AskDocs subreddit submission post threads [6] and on the left-side of the figure we have terms of items we use in our model and an arrow to associate the respective item. A Reddit user can submit a submission post that includes a title and some description (if any) of their question. Once the questions have been approved by the subreddit moderators, anyone can view, upvote, and reply with a comment to the submission post threads. Each submission post thread has an associated upvote score that represents the popularity of the post, as these upvotes are given by the users. These threads link to specific comments answering the submission post. In Figure 4.6 we some sample comments of an example submission post thread from Figure 4.5. This particular submission post thread has a not save for work (NSFW) tag as it depicts some explicit subjects. There are multiple ways a user can view the comments, the top upvoted comments are ranked highest as a default. In this particular example we show two different users "Robotheadbumps" and "CloudSill", both have "Physician" as an author flair, as this gives the reader more detail about the credibility of their answers.

We crawled data from three subreddits, /r/askscience, /r/AskHistorians, and /r/AskDocs from their inception until October 2017 [7]. While these subreddits share the same platform, the communities differ vastly, see Table 4.2. We preprocessed the data by removing uninformative comments and posts with either less than ten characters or containing only URLs or with missing title or author information. We removed users who have posted less than two comments and submissions with three or fewer comments. To handle sparsity, we treated all users with a single comment as "UNK".

Dataset	Created	N	N_e	M	$\lvert a_{m,e}\rvert$	$\lvert w_{m,n}\rvert$
*Docs	07/13	3,334	286	17,342	10,389	53.5
*Science	04/10	73,463	2,195	100,237	70,108	74.0
*Historians	08/11	27,264	296	45,650	30,268	103.4

Table 4.2: Dataset statistics for the subreddit communities. The symbol meaning are as follows: N and M denotes total users and posts respectively; N_e: number of experts; $|a_{m,e}|$: number of posts with at least one expert comment; $|w_{m,n}|$: average comment word length.

In the askscience subreddit, for each submission post, there is an associated flair text denoting the *category* of the post, referred as the *submission flair* that is either Modera-

[5]The dataset can be found at `https://amorale4.github.io/research/`.

[6]As of November 2020.

[7]`praw.readthedocs.io/en/latest/`

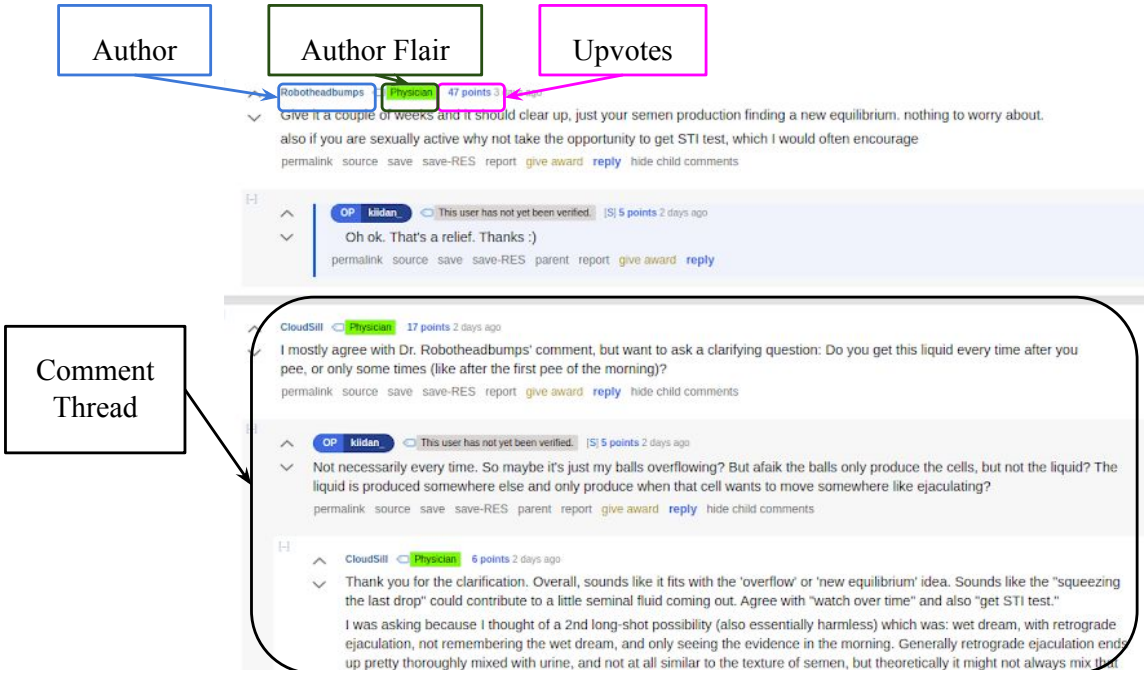


Figure 4.6: Sample replies to the AskDocs submission post thread titled "Thick liquid oozing after pee sessions, signs of prostate leak?". The submission post description is omitted for brevity.

tor added or self-annotated, e.g., Physics, Chemistry, Biology. Similarly, users have *author flairs* attributed next to their username describing their educational background, e.g., Astrophysicist, Bioengineering. Only users verified by the moderator have *author flairs*, and we denote them as experts in the rest of the chapter. AskDocs does not have submission flairs as it is a smaller community. For both subreddits, we observed that around 80% of the users comment on posts from more than two categories. Experts are highly active in the community answering around 60-70% of the posts (Table 4.2). askscience and AskHistorians have significantly higher (Figure 4.8) and more detailed comments ($|w_{m,n}|$ in Table 4.2) per post than AskDocs. Due to the prevalence of a large number of comments, manual curation is very expensive, thus necessitating the need for an automatic tool to infer comments trustworthiness.

4.4.1 Expert Label Collection

To collect the expert labels for the /r/askscience subreddit, we crawled the twenty-six submission post where the community members applied for user flair. In their application they categorized their expertise in one twelve general fields, shown in Table 4.3. As it is possible for users to delete their comment posts, we then matched the users which commented

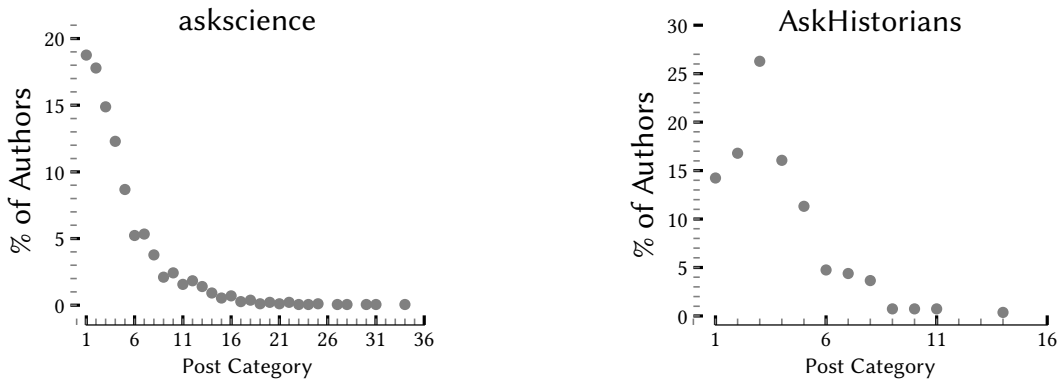


Figure 4.7: Frequency plot of % of authors commenting on the post with unique submission flairs.

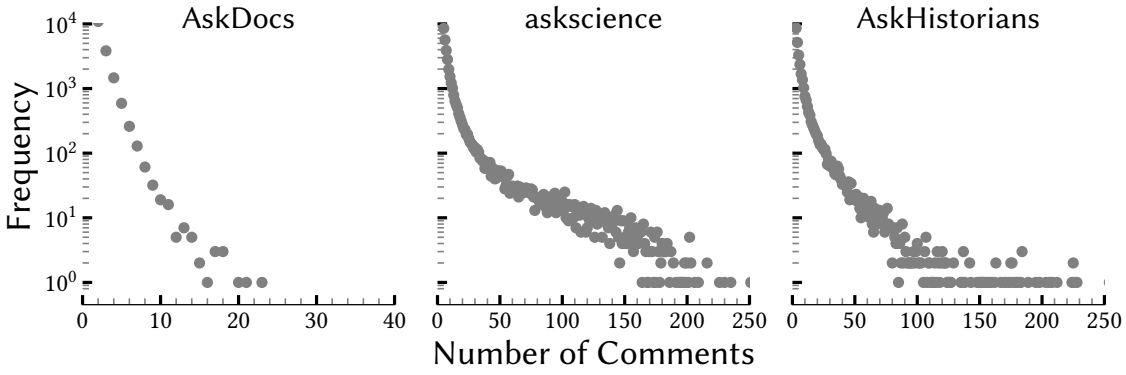


Figure 4.8: Frequency plot (log scale) of number of comments per post for three subreddits. A post on AskDocs tend to have fewer comments than the other two communities.

with their corresponding general flair. While it is possible for users to be assigned multiple specific fields, they can only select one general field. In total we identified 2027 users spanning these twelve general fields. The link to the CrowdQM Reddit dataset can be found here `https://amorale4.github.io/research/`.

4.5 PREDICTING TRUSTWORTHY COMMENTS

In this section, we first discuss our novel dataset, followed by experiments on the learned outputs of our model. In particular, we evaluate the trustworthy comment embeddings on the comment ranking task. While we qualitatively evaluate user reliabilities and word embeddings. For brevity, we focus the qualitative analysis on our largest subreddit, askscience.

General Field	Related/Specific Fields	Number of Experts
Astronomy	Astronomy, Astrophysics, Cosmology, Planetary Formation	145
Biology	Biology, Evolution, Morphology, Ecology, Synthetic Biology, Microbiology, Cellular Biology, Molecular Biology, Paleontology	489
Chemistry	Chemistry, Organic Chemistry, Polymers, Biochemistry	366
Computing	Computing, Artificial Intelligence, Machine Learning, Computability	99
Engineering	Mechanical Engineering, Electrical Engineering, Structural Engineering, Computer Engineering, Aerospace Engineering	121
Mathematics	Mathematics, Statistics, Number Theory, Calculus, Algebra	165
Medicine	Medicine, Oncology, Dentistry, Physiology, Epidemiology, Infectious Disease, Pharmacy, Human Body	171
Neuroscience	Neuroscience, Neurology, Neurochemistry, Cognitive Neuroscience	40
Physics	Theoretical Physics, Experimental Physics, High-energy Physics, Solid-State Physics, Fluid Dynamics, Relativity, Quantum Physics, Plasma Physics	124
Planetary Sciences	Earth Science, Atmospheric Science, Oceanography, Geology	108
Psychology	Psychology, Cognitive Psychology, Developmental Psychology, Abnormal, Social Psychology	95
Social Sciences	Social Science, Political Science, Economics, Archaeology, Anthropology, Linguistics	104

Table 4.3: Distribution of flairs and related/sub-fields for the expertise found on the askscience sub-reddit.

4.5.1 Trustworthy Comment Embedding Analysis

We evaluate latent trustworthy comment learned by our model on a trustworthy comment ranking task. That is, given a submission post, our goal is to rank the posted comment based on their trustworthiness. For this experiment, we treat expert users' comment as the most trustworthy comment of the post. While human judgment would be the most precise; it is also the most challenging to collect. For instance, in askscience we would need experts in over 35 science fields, reading up to 250 comments for a single post. This does not mean that all non-experts give wrong responses, notwithstanding, there could also be unverified users who give high-quality responses in the dataset. Besides, we also report results using the highest upvoted comment as the gold standard. Highest upvoted comments represent community consensus on the most trustworthy response for the post [162]. We rank comments for each post m, in the order of descending cosine similarity between their embedding, $a_{m,n}$, and the learned trustworthy comment embeddings, a^*_m. We then report average Precison@k values over all the posts, where k denotes the position in the output ranked list of comments.

Baselines: We compare our model with state-of-the-art truth discovery methods proposed for continuous and text data and non-aspect version of our model. Note that there is no label information used, so we cannot compare to other supervised CQA models [147, 156, 163] which need this supervision. Our *unsupervised model* is complementary to these approaches, and thus, a rigorous comparison is impossible. Unless stated otherwise, we used the authors' implementation of their model.

Mean Bag of Answers (MBoA) : In this baseline, we represent the trustworthy comment for a post as the mean comment embedding and thus assume uniform user reliability.

CRH : This model is a popular truth discovery-based model for numerical data [134]. CRH minimizes the weighted deviation of the trustworthy comment embedding from the individual comment embeddings with user reliabilities providing the weights. The goal of the optimization problem is to minimize the weighted loss of the aggregation results. For this experiment, we use the average word embeddings of comments as input to the model.

CATD : This model is an extension of CRH that learns a confidence interval over user reliabilities to handle data skewness [164]. For both the above models, we represent each comment as the average word embeddings of its constituent terms.

TrustAnswer [8]: Li et al. [136] modeled semantic similarity between comments by representing each comment with embeddings of its key phrase. Although, they do not model aspect-level user reliability, this model is a special case of our proposed model where we only consider a single topic and assume each user (post) are weighted equally, however they estimate user reliability on the current post and not on user's comments on other posts.

CrowdQM-no-aspect: In this baseline, we condense the user's aspect reliabilities to a single r_n. Similar to our proposed model, however the major difference is each commenter's aspect reliabilities is condensed to a single r_n. This model acts as a control to gauge the performance of our proposed model. We do not compare with other truth discovery methods [149, 150, 151, 152, 153] as CRH and CATD are already shown to outperform them.

Results: Table 4.4a reports the Precision@1 results using expert's comments as the gold standard. MBoA, with uniform source reliability, outperforms the CRH method that estimates reliability for each user separately. Thus, mean embeddings provide a robust representation. We also observe that CrowdQM-no-aspect performs consistently better than TrustAnswer. Note that both approaches do not model aspect level user reliability but use semantic representations of comments. However, while TrustAnswer assigns a single reliability score for each comment, CrowdQM-no-aspect additionally considers the user's familiarity with the post's context (*similarity* function, $s(.)$) to compute her reliability for the post. Finally, CrowdQM consistently outperforms both the models, indicating that aspect modeling is beneficial.

CATD uses a confidence-aware approach to handle data skewness and performs the best among the baselines. This skewness is especially helpful in Reddit as experts are the most active users (Table 4.2); thus, CATD likely assigns them high reliability. Our model achieves competitive precision as CATD for AskDocs. One reason why the model might not work as well as askscience and AskHistorians, is the sparsity in the responses, as there are not many posts which many users jointly comment on, see Section 4.4 and Figure 4.8.

Table 4.4b reports Precision@1 results using community upvoted comments as the gold standard while Figure 4.9a plots the precision values against the size of the output ranked comment list. In general, there is a drop in performance for all models on this metric because it is harder to predict upvotes as they are inherently noisy [148].

TrustAnswer and CrowdQM-no-aspect perform best among the baselines indicating that modeling semantic representation is essential for forums. CrowdQM again consistently outperforms the non-aspect-based models verifying that aspect modeling is needed to identify

[8] We used our own implementation, as there is no code and since this is a special case of CrowdQM.

Model	*Docs	*Science	*Historians
MBoA	0.592	0.633	0.602
CRH [134]	0.585	0.597	0.556
CATD [164]	**0.635**	0.700	0.669
TrustAnswer [136]	0.501	0.657	0.637
CrowdQM-no-aspect	0.509	0.666	0.640
CrowdQM	0.617	**0.734**	**0.753**

(a)

Model	*Docs	*Science	*Historians
MBoA	**0.434**	0.302	0.257
CRH [134]	0.386	0.234	0.183
CATD [164]	0.405	0.291	0.257
TrustAnswer [136]	0.386	0.373	0.449
CrowdQM-no-aspect	0.388	0.368	0.450
CrowdQM	0.426	**0.402**	**0.493**

(b)

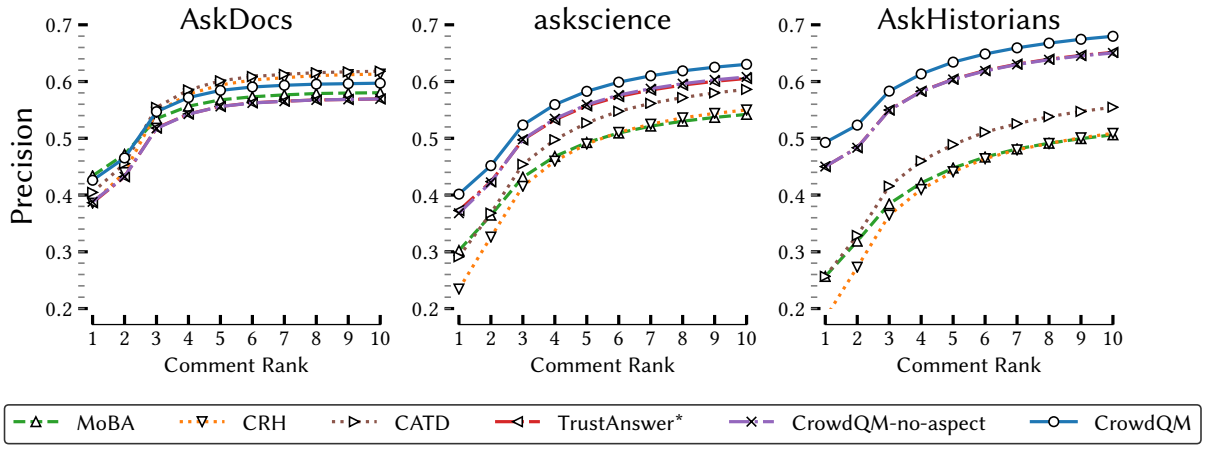
Table 4.4: Precision@1 for all three Ask* subreddits, with (4.4a) the experts' comments and (4.4b) upvotes used to identify trustworthy comments.

trustworthy comment in forums. CrowdQM remains competitive in the smaller AskDocs dataset, where the best performing model is MoBA. Thus, for AskDocs, comment summarizing all the other comments tends to get highest votes.

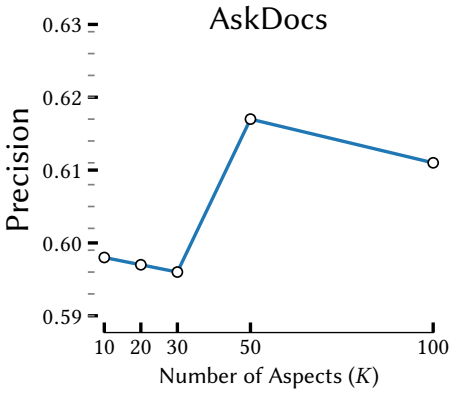
Parameter Sensitivity In Figure 4.9b, we plot our model's precision with varying number of aspects. Although there is an optimal range around 50 aspects, the precision remains relatively stable indicating that our model is not sensitive to aspects. We also observed similar results for the other datasets. We also did similar analysis with β and did not find any significant changes to the Precision.

4.5.2 Model Convergence

In Figure 4.10, we plot the objective function score at each iteration, for our model CrowdQM for the three datasets. On all three datasets, the model converges after only about six iterations indicating our model is quick to approximate a solution. In general, the computational complexity is $O(|\mathcal{V}|NM)$ for a single iteration. However, our implementation leverages the data sparsity in the comment-word usage and user-submissions posts.



(a)



(b)

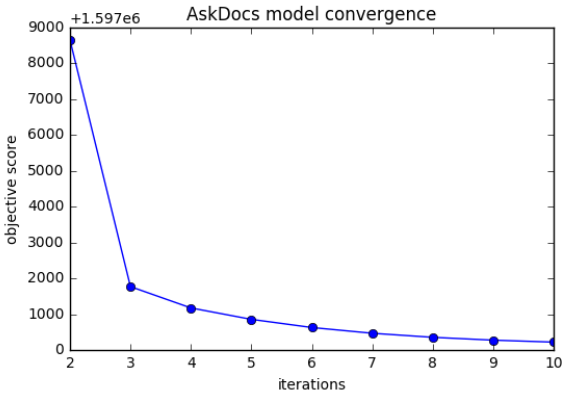
Figure 4.9: Precision of our model (4.9a) vs. comment rank computed by user's upvotes and (4.9b) vs. number of aspects. Our model outperforms the baselines for askscience and AskHistorians while performs similarly for AskDocs. Value of K does not have much impact on the precision value.

4.6 CROWDQM QUALITATIVE ANALYSIS

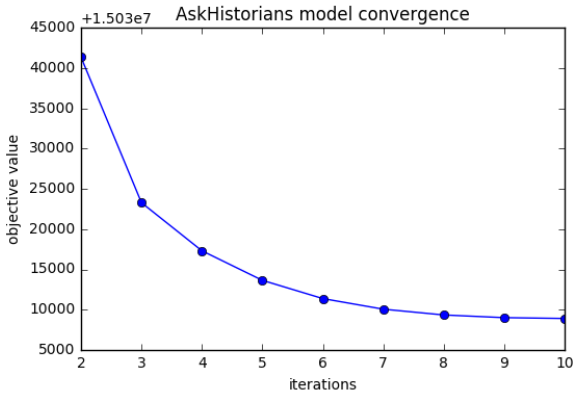
In this section, we report qualitative analysis of user-aspect reliabilities $\{r_n\}$ and word embeddings $\{v_w\}$ learned by our proposed CrowdQM model. For brevity, we focus our analysis on our largest subreddit, askscience.

4.6.1 Aspect Reliability Analysis

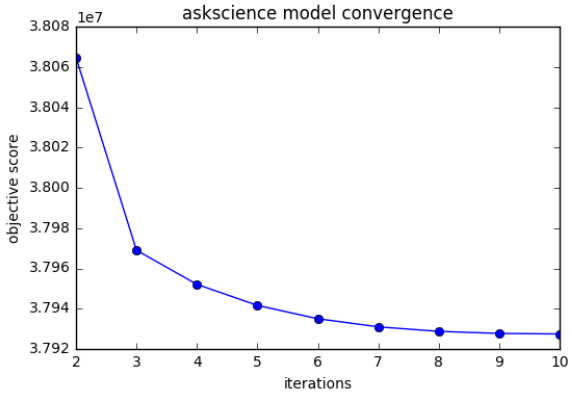
We evaluate learned user reliabilities for users who commented on a post with a *submission flair*. Note that a submission flair is manually curated and denotes post's category, and we do not use this information in our model. Specifically, for each post m, we compute the *user-*



(a) AskDocs



(b) AskHistorians



(c) askscience

Figure 4.10: CrowdQM model convergence for AskDocs, AskHistorians, and askscience respectively.

post reliability score R_m^n for every user n who commented on the post. We then ranked these scores for each category and report top *author flairs* for few categories in Table 4.5. The top-performing *author flairs* for each category are experts for that domain. For instance, for the Computing category highly reliable users have author flairs like Software Engineering and Machine Learning, while for Linguistics authors with flairs Hispanic Sociolinguistics and Language Documentation rank high. These results align with our hypothesis that in-domain experts should have higher reliabilities. We also observe that out of domain authors with flairs like Comparative Political Behavior and Nanostructured Materials in the Linguistic category. This diversity could be due to the interdisciplinary nature of that domain. Thus, our model can also be used by moderators of the community forum to identify and recommend potential reliable users to respond to new submission posts of a particular category.

Post Category: Computing	Post Category:Linguistics
Embedded Systems ; Software Engineering ; Robotics	Linguistics ; Hispanic Sociolinguistics
Computer Science	Comparative Political Behaviour
Quantum Optics ; Singular Optics	Historical Linguistics ; Language Documentation
Robotics ; Machine Learning ; Computer Vision ; Manipulators	Linguistics ; Hispanic Sociolinguistics
Computer Science	Historical Linguistics ; Language Documentation
High Performance Computing ; Network Modeling and Simulation	Cognitive Modeling
Biomechanical Engineering ; Biomaterials	Nanostructured Materials ; Heterogeneous Catalysis
Post Category: Archaeology	**Post Category: Medicine**
Archaeology ; Maya Stone Tools ; Geoscience	Infectious Diseases ; Pulmonary Immunology
Global Health ; Tropical Medicine	Biomedical Engineeering ; Biomechanics ; Biomaterials
Control ; Robotics Engineering ; Industrial Robotics	Pediatric Neurology
Archaeology ; Collapse of Complex Societies	Anesthesiology ; Post-Operative Pain ; Traumatic Brain Injuries
Archaeology ; Archaeometallurgy	Molecular Biology ; Musculoskeletal Research
Criminal Justice	Immunology ; Immune Regulation ; Infectious Diseases
Computational and Evolutionary Archaeology	Molecular Biochemistry ; DNA Damage Repair
Post Category: Biology	**Post Category: Psychology**
Animal Cognition	Clinical Psychology ; Psychotherapy ; Behavior Analysis
Cell and Developmental Biology	International Relations ; Comparative Politics
Biochemistry ; Molecular Biology ; Enzymology	Neuropsychology
Genetics ; Cell biology ; Bioengineering	Psychology ; PTSD, Trauma, and Resilience
Computational Physics ; Biological Physics	Cognitive Neuroscience ; Neuroimaging ; fMRI
Aquatic Ecology and Evolution ; Active Acoustics	Psychology ; Legal psychology ; Eyewitness testimonies
Genomic Instability ; Cancer Development	Experimental Psychology ; Social Cognition and Statistics

Table 4.5: Top author flairs with their corresponding post categories according to user-post reliability score.

To further analyze the user-aspect reliability, we identify the most important aspect for each post category. We correlate the user *karma*, computed for each post category, with their reliability score in each k aspect, $r_n^{(k)}$. For this experiment, category-specific karma is given by the average upvotes the user's comments have received per category. Users with high karma value are deemed reliable by the community for that category. We identify aspects for each category using the highest correlation value of user reliability and karma value. Table 4.11 list the top words of the correlated aspect for some categories. The identified aspects words are topically relevant thus our model can associate user aspect reliability coherently. It is interesting to note that, the aspects themselves tend to encompass several themes, for example, in the Health category, the themes are software and health.

4.6.2 Word Embedding Analysis

The CrowdQM model updates word embeddings to better model semantic meaning of the comments. For each category, we identify the frequent terms and find its most similar keywords using cosine distance between the learned word embeddings.

The left column for each term in Table 4.6 are the most similar terms returned by the initial embeddings while the right column reports the results from updated embeddings $\{v_\omega\}$

Liquid		Cancer		Quantum		Life	
Initial	CrowdQM	Initial	CrowdQM	Initial	CrowdQM	Initial	CrowdQM
unimaginably	gas	mg	disease	search results	model	molaison	species
bigger so	chemical	curie	white	sis	energy	around	natural
two lenses	solid	wobbly	cell	shallower water	particle	machos	nature
orbiting around	air	subject	food	starts rolling	mechanics	brain	production
fire itself	material	"yes" then	complete	antimatter galaxies	mathematical	"dark" matter	size

Table 4.6: Similar words using embeddings learned using CrowdQM for askscience.

from our CrowdQM model. We observe that there is a lot of noise in words returned by the initial model as they are just co-occurrence based while words returned by our model are semantically similar and describe similar concepts. This improvement is because our model updates word embeddings in a trust aware manner such that they are similar to terms used in responses from reliable users.

4.7 CROWDQM-BASED FEATURE CONSTRUCTION

In this section we describe how we can leverage the CrowdQM model to generate features for text prediction. There are many possible feature constructions from the latent aspects from the CrowdQM model, while some may be more useful than others it depends on the context for which they are used. For example, in the context of trustworthy comment discover we used a^*_m as a feature. One feature which is particularly useful for topic related categorizations is source-aspect reliability features, r_n. However, to fully take advantage of these features it is best to take the user familiarity in conjunction with these features.

$$R^{(k)}_n = r^{(k)}_n \cdot u^{(k)}_n \tag{4.11}$$

To show the utility of these features we focus on expert categorization, for the askscience subreddit.

We establish a baseline for the task in the CrowdQM Reddit dataset, for expert categorization. As features we compare unigram-based features, which are simply the term frequencies. We also compare topic-based feature which we derive from applying LDA to the corpus.

In Figure 4.12 we show the precision of these features, in general the CrowdQM-based features outperform the topic and unigram-based features except in the Computing and Social Sciences categories, which unigram gets a precision of 1.0. In Figure 4.13 we show the recall of the three features and as we can see in all but the two categories the CrowdQM features outperform. In Figure 4.14, we show the F1-Score for each category, while CrowdQM outperforms the other two types of features there is still a substantial room for improvement.

4.8 TRUTH DISCOVERY AND COMMUNITY QUESTION ANSWERING RELATED WORK

Our work in this chapter is related to several themes of research, including truth discovery and question answering.

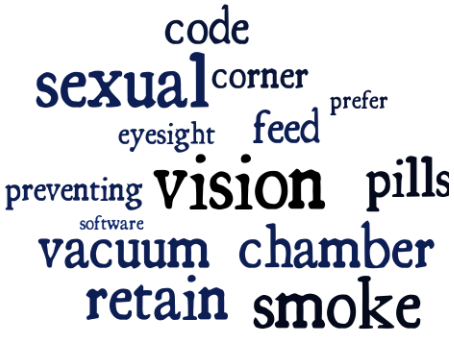
In SemEval 2017 on Community Question Answering (CQA), [147] developed a task with the following end-application goal: given a new question, the system should automatically recommend useful related answers. SemEval 2019 further extends this line of work by proposing fact checking in community question answering [158]. Typically, CQA is framed as a classification problem to predict correct responses for a post. CQARank leverages voting information as well as user history and estimates user interests and expertise on different topics [135]. [157] also look at the relationship between the answers, measuring textual and structural similarities between them to classify useful and relevant answers. These are supervised approaches and thus need a large amount of labeled training data [163, 165, 166]. Our goal is different as we want to identify the most trustworthy response within a post which subsumes relevance to the question [167].

Truth discovery has attracted much attention recently. Different approaches have been proposed to address different scenarios [155, 168, 169, 170, 171, 172]. Many truth discovery approaches are tailored to categorical data and thus assume there is a single objective truth that can be derived from the claims of different sources [173]. Faitcrowd [174] assumes an objective truth in the answer set and uses a probabilistic generative model to perform fine-grained truth discovery. It jointly models the generation of questions and answers to estimate the source reliability and correct answer. On the other hand, [154] propose trustworthy *opinion* discovery where the true value of an entity is modeled as a random variable with a probability density function instead of a single value.

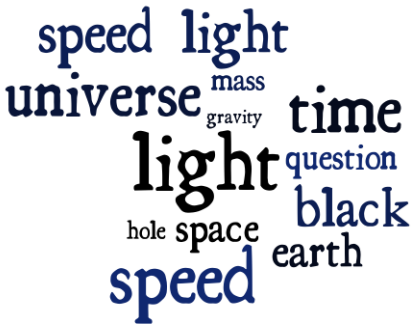
Some truth discovery approaches also leverage text data to identify correct responses better. [136] proposed a model for capturing semantic meanings of crowd provided diagnosis in a Chinese medical forum. [155] proposed a Bayesian approach to capture the multifactorial property of text answers and used semantic representations of keywords to mitigate the diversity of words in answers. These approaches only use certain keywords for each answer and are thus, limited in their scope. To the best of our knowledge, there has been no work that models fine-grained user reliability with semantic representations of the text to discover trustworthy comments from community responses.

4.9 SUMMARY

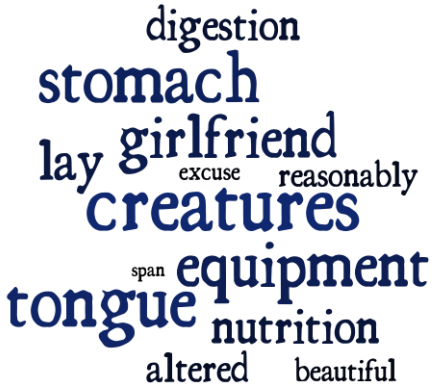
In this section we have proposed source reliability aspect features as a form of model-based features. We proposed an unsupervised model to learn a trustworthy comment embedding from all the given comments for each post in a discussion forum. The learned embedding can be further used to rank the comments for that post. We explored Reddit, a novel community discussion forum dataset for this task. Reddit is challenging as posts typically receive many responses from a diverse set of users and each user engages in a wide range of topics. Our model estimates aspect-level user reliability and semantic representation of each comment simultaneously. Experiments show that modeling aspect level user reliability improves the prediction performance compared to the non-aspect version of our model. We also show that the estimated user-post reliability can be used to identify trustworthy users for post categories. We applied CrowdQM-based features for an expert prediction task and showed the utility of these features for this categorization task.



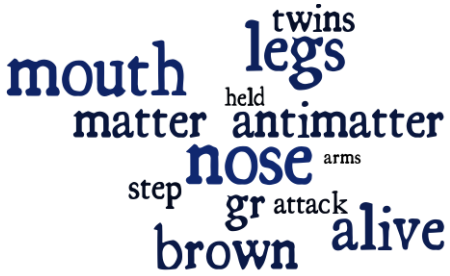
(a) Health



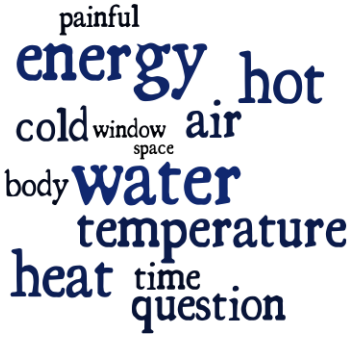
(b) Cosmos



(c) Diabetes



(d) Oceanography



(e) Astronomoy

Figure 4.11: Top words for highly correlated aspects between user reliability and user karma.

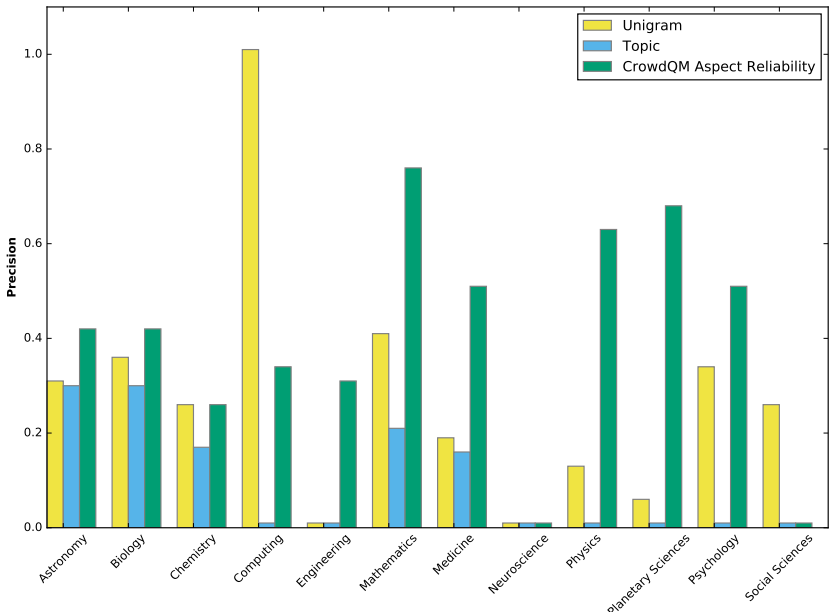


Figure 4.12: Precision of three different feature types for the expert classification prediction task. The X-axis denote the corresponding Field of study.

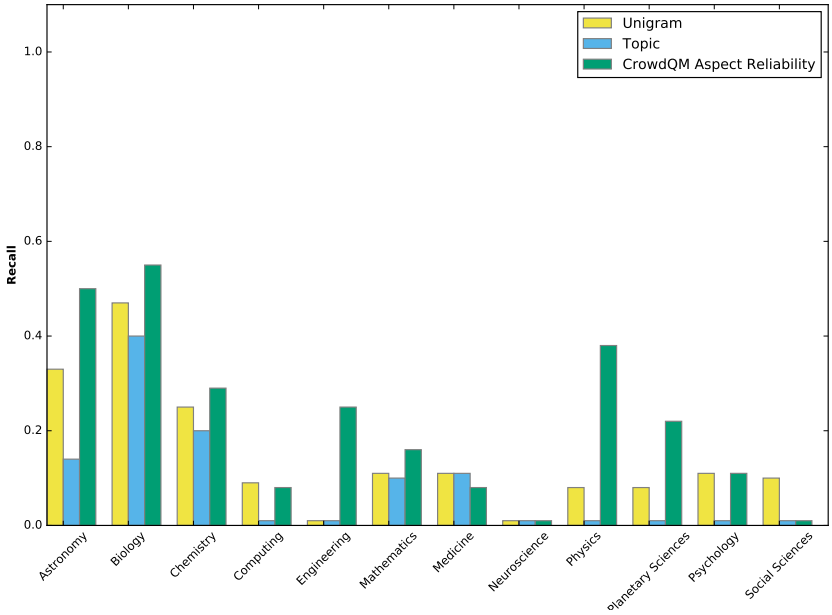


Figure 4.13: Recall of three different feature types for the expert classification prediction task. The X-axis denote the corresponding Field of study.

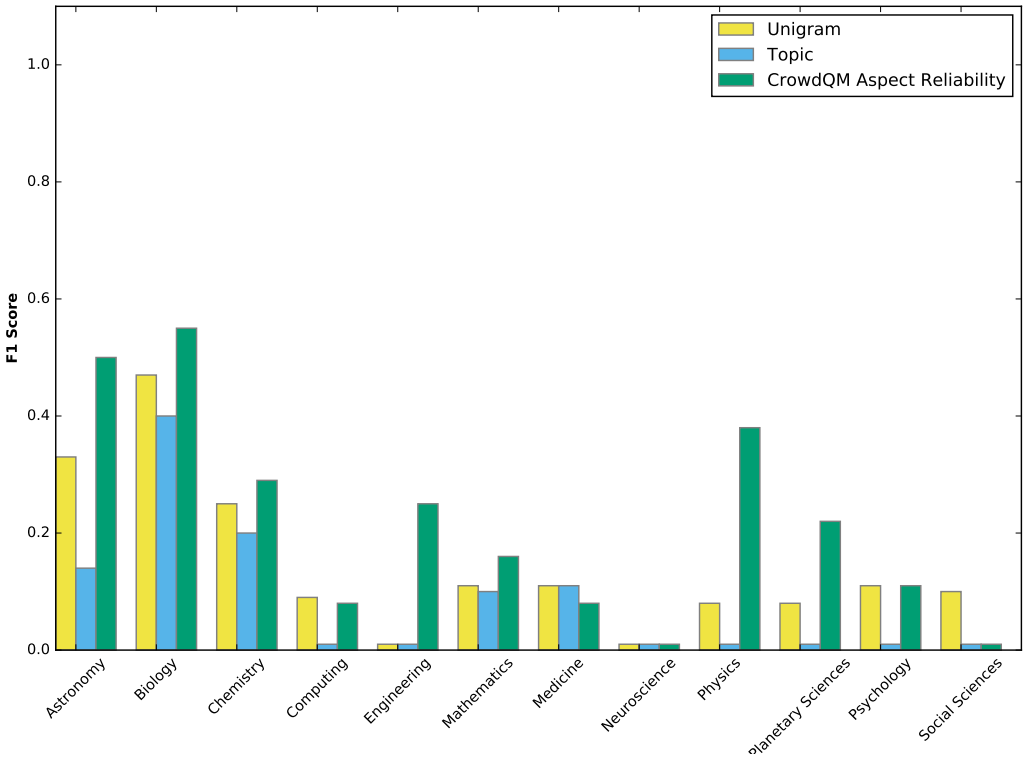


Figure 4.14: F1 Score of three different feature types for the expert classification prediction task. The X-axis denote the corresponding Field of study.

CHAPTER 5: MULTI-VIEW ATTRIBUTE FEATURES

In Chapter 3 and Chapter 4 we have presented two new models for modeling semantic incoherence and reliability of text content respectively. These models can be used to construct useful features that are effective for several interesting applications. In this chapter we present multi-view feature construction which leverage any model-based features to reconstruct features from different perspectives of the data.

5.1 TOPIC FEATURES FOR TWEET-BASED PREDICTION

In this chapter we focus on text-based features and multi-view attribute features for the purpose of making predictions from text data. In particular, we focus on the prediction of new diagnosis rates of sexually transmitted infections (STIs) in particular locations based on social media posts from users in those locations. *Multi-View Attribute Feature Construction* is a topic modelling framework for topic feature construction on social media text data which leverages attributes of social media. This framework allows us to include meta-data, as attributes, for construction of coherent topic-based features.

While topic models provide a feature for prediction, it is not always clear what document representation of our text data, e.g., tweet messages, should be used in a prediction task (e.g., predicting sexually transmitted infections (STIs) new diagnosis rates). One *naive document representation* might be to pool all messages into a document belonging to a particular location and later infer corresponding features given a new location. However, the resulting pooled documents might not be topically coherent, alternatively it is also possible to consider each message as an individual document, thus having multiple attribute documents for a given location. In particular, the misalignment of the text data, e.g. tweets, and the prediction outcome, e.g. STIs diagnosis rates at the county-level, poses a challenge for feature construction and thus a framework for multi-view attribute feature construction is necessary for this application.

We develop a novel general framework for constructing multi-attribute topic features using multi-views of the social media text data defined according to meta-data attributes and study their effectiveness for a text-based prediction task. We show the relationship between multi-view attribute features construction and model-based feature construction. Furthermore, we study multiple weighting strategies and attributes for multi-view attribute feature construction to align text-based features and prediction outcomes. We evaluate the proposed method on a Twitter corpus of over 100 million tweets collected over a seven-year period in

2009-2015 to predict human immunodeficiency virus (HIV) new diagnosis and other STIs new diagnosis in the United States at the zip code-level and county-level resolutions. The results show that feature representations based on attributes such as authors, locations, and hashtags are generally more effective than the conventional topic feature representation.

5.1.1 Multi-view Attribute Features for STI Prediction

The abundance, and ubiquity, of social media data and the live-stream reporting of events make social media data especially valuable for prediction tasks in many application domains (e.g., security [175] and financial domain [176]). As an instance of "big data," social media data has several unique properties: 1) They are massive, cover a wide range of topics, and represent opinions from a diverse population, thus they contain valuable information relevant to many big data applications. They are especially useful for predicting people's attitude, opinions, and preferences, but can also be used as a basis for predicting many other interesting variables such as stock prices, election results, product trends, and public policy responses. 2) They are often the first source to find a report of an event, thus they are especially useful for making real-time predictions of interesting variables.

Social media data provide real-time signals about various events in the world and thus can be potentially used to make predictions in many applications such as tracking and monitoring diseases to improve disease case reporting for modern disease surveillance. The effectiveness of social media-based prediction highly depends on whether we can construct effective content-based features based on social media text data. Features constructed based on topics learned using a topic model are very attractive due to their expressiveness in semantic representation and accommodation of inexact matching of semantically related words.

While there are many applications of social media, using social media for prediction is especially important because it can directly help optimize decision making and can also be combined with other non-text data in a predictive model. As in many cases of text-prediction applications, the accuracy of prediction, based on social media, would highly depend on whether we can construct effective features using the social media data, thus, how to construct effective features is an extremely important research question in social media mining. While commonly used features such as bag-of-words representation are often effective, they have clear limitations. First, many words are ambiguous. Second, the same concept may be expressed using different terms, causing a mismatch. These limitations can be addressed by using topics as features where a topic is defined as a word distribution (i.e., a unigram language model) since such a topic feature representation would address both the

ambiguity problem and the vocabulary variation problem [177].

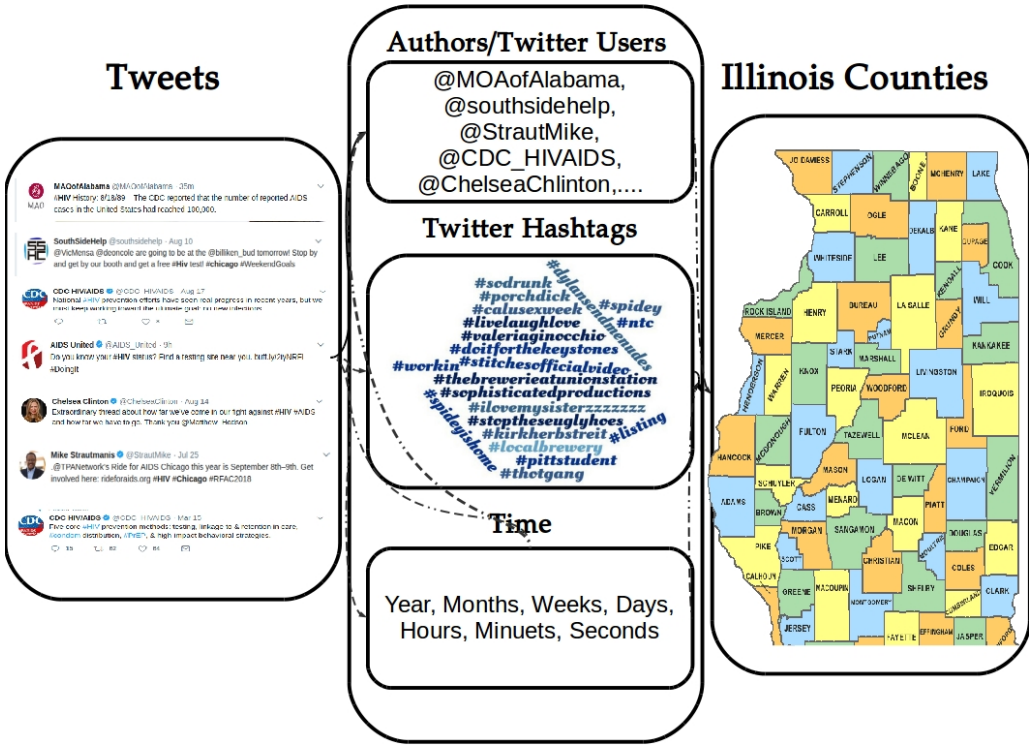


Figure 5.1: HIV new diagnosis prediction from tweets using multi-view attributes to construct features. On the left-most box are the basic text units, tweet and on the right-most box are the prediction outcome we are interested in. In between are different views of the tweets defined based on various (metadata) attributes (i.e. authors, hashtags, and time), which allowed us to generate topic features from multiple perspectives for predicting the outcomes. The dashed arrows here represent a partition relationship which we can define to construct our multi-attribute topic features.

For this reason, statistical topic modeling, in particular Latent Dirichlet Allocation (LDA) [32], is often applied to social media text data for content-based feature construction for predicting health related outcomes and other applications. Though promising, a straightforward application of topic modeling to tweets tends to be not very effective. Specifically, Twitter, as a source of information, is limited by the message length at 140 characters[1], which restricts the types of content-based features used.

Direct application of a topic model such as LDA [32] to tweets has been shown to produce low-quality topics and thus it is crucial to pool tweets to create coherent documents [178, 179]. However, it remains an open challenge how to pool the tweets and how to construct

[1]As of September 2017, Twitter has extended the length limit to included 280 characters for some select users.

effective topic-based features to represent tweets in a prediction task, particularly how to determine values of topic features and how to weigh topics for a prediction task.

In this chapter we propose a general framework for constructing topic-based features on social media text data from multiple views that correspond to different ways to pool social media text such as tweets. Those views are defined based on meaningful meta data such as authors, location, and time, each leading to a different, but coherent way of partitioning and pooling text data, and thus enabling generation of coherent topics representing the text data from a different perspective.

With the proposed approach, we would be able to generate multiple versions of topic models from the text data, each corresponding to a view defined by an attribute such as an author or location. The multiple attributes allow us to represent text data flexibly in different perspectives (views), which is needed for different prediction tasks and provides better discrimination than topics constructed in a conventional way.

In a typical prediction task, it would be naive to collect all the social media text associated with a prediction instance (e.g., a county in the case of predicting HIV rates of different counties) to form a pooled document representation and derive a feature representation for such a document and use in a machine learning predictive model. This method of pooling documents results in incoherent documents and thus suboptimal topic features. Alternatively, with multi-attribute topic features, each document representation is often decomposed into multiple sub-documents corresponding to different attribute values. Thus, we also need to further study how to combine the topic features obtained from multiple "subdocuments" of the pooled document. To this end, we propose multiple weighting strategies for combing topic features.

The basic idea of the proposed multi-view attribute features in the context of predicting HIV rates of each county is illustrated in Figure 5.1.

We evaluate the proposed multi-view attribute topic features using a case study of predicting the HIV rates using tweets, which has important applications. In a recent report by the Center for Disease Control and Prevention (CDC) in 2016 they found in the U.S. 1.59 million cases of Chlamydia, 468,514 cases of Gonorrhea and 27,814 cases of Syphilis, a 4.7%, 18.5% and 17.6%, respectively, increase from 2015 [180]. Monitoring the prevalence of STIs and HIV is essential for timely reportage for infection prevention and control and cost planning. Social-media, e.g., Twitter, allows for a platform to mine health related markers (e.g., discussion of health-related topics), and studies have shown some potential for tracking health related outcomes such as HIV [73, 74, 75, 86, 181]. We thus chose to evaluate the proposed feature construction method with the task of predicting STIs based on tweets.

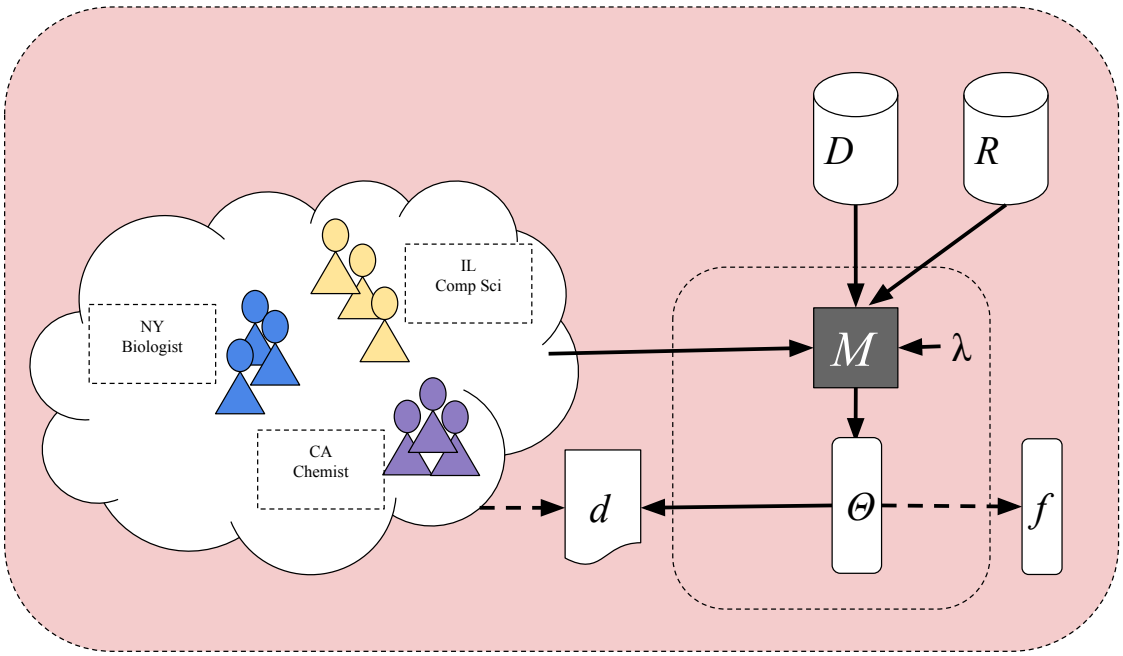


Figure 5.2: Multi-view attributes in the model-based feature construction framework. The cloud represents multiple attributes of the data such as location, profession, and expertise. These attributes can be used to cluster the data and provide multiple views on the feature construction process.

5.1.2 Multi-View Attribute Feature Set for Text Data

In Chapter 3, we showed that model-based feature construction can leverage reference corpus to create differential semantic features. In Chapter 4, we described a model that leveraged sources of text to model source aspect reliabilities. In this chapter we leverage the meta-data which naturally occurs in social media to construct multi-view attribute features. In Figure 5.2, we show the multi-view context of meta-data, which can include user location, user expertise, as well as temporal information, or other meta-data information in social media networks.

5.2 FEATURES IN SOCIAL MEDIA PREDICTION

Text-based prediction can be described as predicting the value of an interesting (dependent) variable (e.g., HIV rates of a county) based on the text data associated with the variable (e.g., all the tweets produced by people from a county). Such a prediction task is representative of "big data" applications in general, where the data is leveraged to make a prediction of an interesting variable, which further helps support and optimize decision

making.

A text-based prediction problem is generally solved by using supervised machine learning which can leverage labeled training data. The general idea is to generate features from the relevant text and hypothesize that the target variable value is a function of those features which has parameters to control how the features should be combined to produce a prediction score. The parameters can then be optimized based on a training set to minimize the prediction error on the training data. The accuracy of prediction depends heavily on the features constructed to represent the text data.

The most commonly used features for representing text data are lexical features such as words, n-grams, and phrases, or some mixture of words and syntactic information such as POS tags [177]; since words can be regarded as human-generated primitive features, they are usually quite effective, leading to the widespread adoption of the "bag-of-words" representation.

However, while lexical features are often sufficiently effective for some tasks where the target variable to be predicted is closely related to the surface lexical features (e.g., in topic categorization of text data or sentiment analysis), they have some notable deficiencies, mostly due to the ambiguity of words and the lack of expressive power when we use one word or a few words as a feature to represent text.

To improve over such a simple representation, topic modeling techniques (notably Latent Dirichlet Allocation (LDA) [32]) have been applied to text data to generate topics that can be used as features. A topic is a word distribution (also called a unigram language model) with high probabilities assigned to important words characterizing a topic. A word distribution is far more flexible and more powerful than a word or a few words when it comes to representing text data, making topics potentially better features than simple lexical features such as n-grams. Moreover, topic features can also be combined with other features such as n-grams to provide supplementary perspectives of representation.

Topics can be learned from text data in an unsupervised way by using a topic model such as LDA [32]. Specifically, given a set of text documents, topic models, such as LDA, can be used to generate two useful outputs $\mathcal{T} = \{\Theta, \Phi\}$, where Φ is a set of topics, each represented as a word distribution, and Θ is a topic distribution for each document indicating the coverage of each topic in the document.

In Figure 5.3, we show the plate notation for LDA. Note α, and β are hyper parameters and the priors for the Dirichlet distributions.

Normally, when we are concerned with a prediction task based on each document, Φ can be used as word clusters representing the features and Θ can directly provide the weights of all the features for each document. However, such a conventional approach is generally

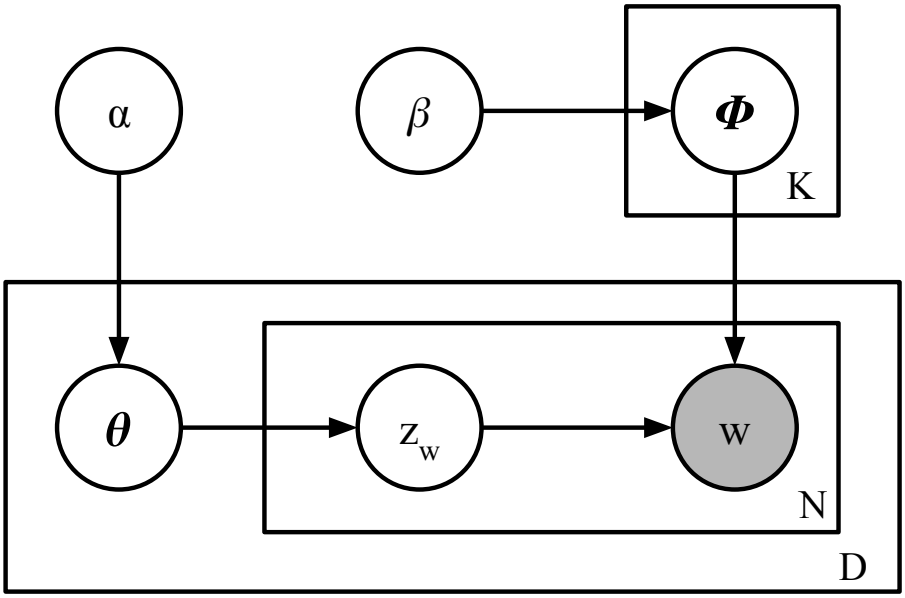


Figure 5.3: The graphical model of LDA in plate notation, where Θ and Φ are the document-topic distribution and topic-word distributions respectively.

inappropriate for many prediction tasks that are not based on a well-defined single document, which include most prediction applications using social media where we generally have to pool multiple tweets together to form a "document" for prediction. For example, in our prediction task of predicting HIV rates in different counties, we would need to pool all the tweets in a county as a "pseudo document." How to learn topics from the data and how to assign values to topic features in such a scenario where we do not have a naturally well-defined document is an open challenge that has not been well addressed in the literature.

In general, we address the following two questions:

- How should we form documents for running the topic model (e.g., one tweet as a document vs. all the combined tweets for each prediction instance)?
- Once we obtain the topics, how do we compute the weights of those topics for each prediction instance (the topic model can no longer give us such weights directly)?

5.3 ATTRIBUTE FEATURE SET FOR TEXT DATA

The general idea of multi-view feature construction is representing multiple views of data, via attributes, as text documents from which we can construct features. Before we give a

definition of multi-view attribute features, we first give some context on attribute feature construction.

In the bag-of-words, a *bag*, or multi-set, of words is defined over the set of all possible terms, i.e., a vocabulary $\mathcal{V}$, and a multiplicity function f defined over V. The multiplicity function f gives the count of $w \in \mathcal{V}$ in the set [182], for example, if $\mathcal{V}$ is the days of a week then

$$\begin{aligned} f(\{Monday, Monday, Monday, Tuesday, Saturday, Saturday\}) \\ \rightarrow \{(Monday, 3), (Tuesday, 1), (Saturday, 2)\} \end{aligned} \tag{5.1}$$

The definition of a bag is not solely limited to unigrams, we can extend this definition to include any arbitrary sequence of characters, or strings. Let A be the set of strings which can be formed, by concatenating the words in $\mathcal{V}$.[2] Similar to how we define the bag-of-words, we can construct a bag-of-strings from A, and the multiplicity function f.

A multi-view feature set is akin to a bag-of-strings, where the multiplicity function is replaced by a feature measure. Recall from equation 2.1, a *feature measure* takes any piece of text and maps it to a real valued representation, e.g., scalar, vector, or tensor. As an example, consider the following strings,

$$A_1 = \textit{What a fine day, for science.} \tag{5.2}$$

$$A_2 = \textit{Today was a fine day for science.} \tag{5.3}$$

$$A_3 = A_1 + A_2 = \textit{What a fine day, for science. Today was a fine day for science.} \tag{5.4}$$

For example, we can define a *feature measure* m as follows, let m be the number of vowels in the string, then $m(A_1) = 9$ and $m(A_2) = 11$. Note that for the concatenation of these two strings $m(A_3) = m(A_1) + m(A_2) = 20$. The feature measure is not limited to a scalar, the feature measure can be a map to a vector or map to some multi-dimensional vector. The idea of multi-view features for text documents is to represent a text document as partitions of sub-documents i.e., if A_3 is the original document then A_1 and A_2 are sub-documents that form a 2-set partition for A_3. A_3 itself can be considered a 1-set partition for A_3, we call this the *identity* partition of A_3. We call the partitioning dimensions *attributes* of the text data. In the above example, we can form a 2-set partition of A_3 via its sentence attribute.

The problem of model-base feature construction can be broken down to defining the fol-

[2]Note, by concatenation we mean to "pool" two terms s.t. $w_1, w_2 \in \mathcal{V}$ then $w_1\ w_2 \in A$.

lowing function

$$\mathcal{M}_\Lambda(A_3) = \mathcal{M}_\Lambda(A_1) + \mathcal{M}_\Lambda(A_2). \tag{5.5}$$

This framework for constructing the multi-view features by leveraging various attributes, can be applied to the meta-data attributes that are naturally available in most social media data. For clarity, we often use tweets as examples to illustrate an idea or technique, but the idea and technique are usually general and can be applied to any social media data.

5.4 MULTI-VIEW ATTRIBUTE FEATURE CONSTRUCTION

A main challenge in topic weighting is that in many topic modelling applications, there is often a misalignment of the text document representation and the outcome variables, e.g., a tweet message vs zip code-quarterly HIV new diagnosis rate. We call this the *outcome misalignment problem*. While pooling the data may remedy this issue, those pooled documents may not be topically coherent. The pooled documents can unintentionally introduce population-based feature biases and hurt the prediction performance.

To address this challenge, we propose a general framework for computing multi-attribute topic representations (called multi-view attribute features), which can preserve topically coherent documents and reduce those inherent population biases. Our key idea is to leverage the naturally available attributes in social media (e.g., authors, location and hashtags) to obtain multiple views of the tweets, each being semantically coherent, and thus enrich the feature representation to increase the chance of obtaining effective topic features for a given prediction task.

We observe the outcome misalignment problem in Figure 5.1 where the outcomes are associated to counties while the text data is at the tweet message level. We can naively pool these tweets to create a *pseudo-document* representation for each county, however these pooled documents may not be topically coherent. We call this a pseudo-document representation as this document is artificially created by pooling tweet messages which share the same location. Alternatively, for each county pseudo-document we can partition by the attributes such as authors, hashtags, or the timestamp in which the tweets were created. Just like the authors partition the counties document set, so we can say that the tweets themselves partition the author document sets, in other words in the dataset we find tweets, written by different authors in different locations. Generally, we can use as many attributes as available, i.e., we can partition the tweets by authors by time of day, but for clarity we describe our model for a single attribute even though partitioning can be done by combinations of

multiple attributes.

Let d be a document in our document collection $d \in \mathcal{D}$. To construct features for prediction, our text document representation, d, needs to be at the same granularity as the predicting data (outcome variable). For example, if we want to predict the HIV rates at county level, each document d would be all the tweets written by people in a particular county. Such an ad hoc combination of all the tweets makes d incoherent, thus using d as a unit for running a topic modeling would be problematic since there will be noisy co-occurrences that may be picked up by the topic model.

Fortunately, it is often the case that in social media we have more detailed information about these documents available, e.g. meta-data such as message authorship information, which can help us develop better topical features. Specifically, the document $d = \{a_1, a_2, ..., a_{M_d}\}$, can be viewed as a collection of some attribute a, i.e., a view of the data under the lens of a; in other words, we say a partitions d. Here M_d denotes the number of partitions given by the particular attribute for document d, see Figure 5.4. For example, if the attribute a is authors, then the document may be partitioned by the messages which are written by different authors. The tweets that have the same attribute value form a sub-document that we refer to as an *attribute document*, and is denoted by a_i. Thus, if there are 1,000 authors in total, we would have $d = \{a_1, ..., a_{1000}\}$, where a_i is all the tweets in document d that are written by author i, and in effect, we partitioned all the tweets in a particular county into 1,000 subsets (i.e., 1,000 attribute documents), each corresponding to the tweets written by a particular author in that county. If we use another attribute (e.g., time), we would have another way (i.e., another view) to partition the same document d.

Given a particular attribute a, we can then use all its corresponding attribute documents in the entire dataset as text units (i.e., as a "document") to run a topic model and generate topics and topic distributions for all the attribute documents, which we denote by $\mathcal{T}_a = \{\Theta^{(a)}, \Phi^{(a)}\}$ with $\Theta^{(a)}$ being the topic distributions and $\Phi^{(a)}$ being the word distributions for all the topics discovered.

Thus, for attribute (view) a, we can take all the topics in $\Phi^{(a)}$ each as a feature, and compute the weight of feature k (i.e., topic k) in the feature representation for document d as follows:

$$\theta_{dk} = P(z = k|d) \tag{5.6}$$

where z is a latent variable indicating the topic in document d. Since an attribute forms a

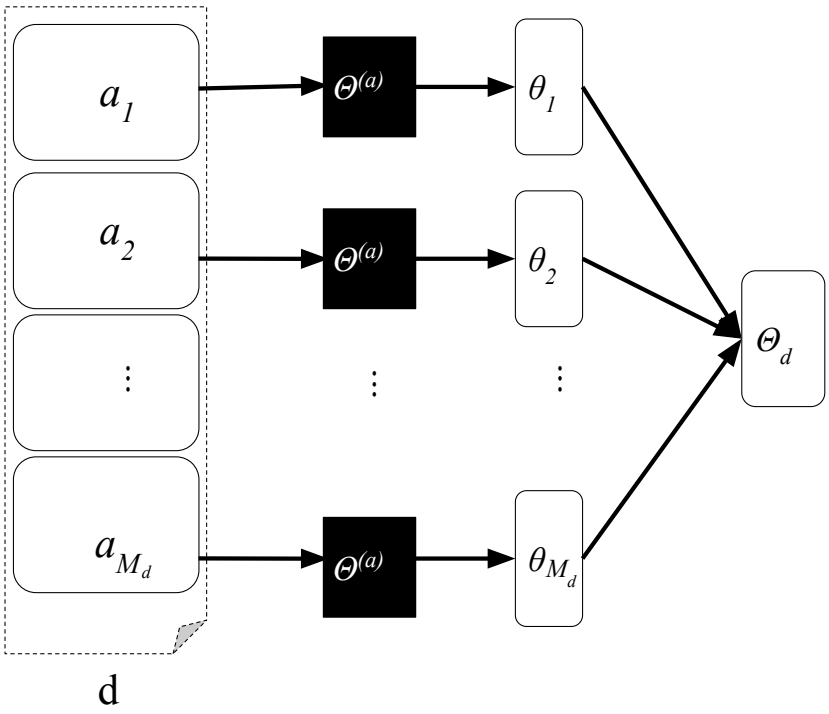


Figure 5.4: On the left we have the document partitioned by n attributes. The black box model is an LDA topic model for the respective attribute, for which we can induce attribute topic distributions, finally we use these to infer the document-topic distribution of the original document.

partition, we can marginalize over the attribute documents,

$$P(z = k|d) = \sum_{a_i \in d} P(z = k|a_i, d)P(a_i|d) \tag{5.7}$$

where $P(z = k|a_i, d)$ is the topic weight for a partition of d by attribute value a_i that we can directly obtain from $\Theta^{(a)}$. The last term $P(a_i|d)$ signifies the weight of attribute a_i in d, and we will discuss how to set this weight below.

In Figure 5.4 show how we can reconstruct the document topic distribution θ_d from the inferred attribute topic distributions $\theta_i^{(a)} = P(z|a_i, d)$.

5.4.1 Attribute Feature Weighting

First, we can consider *balanced topic weight* (BTW), which is defined as,

$$P(a_i|d) = \frac{1}{M_d} \tag{5.8}$$

In such a weighting method, we view every distinct value of attribute a as equally important, thus avoiding any bias we might have due to non-uniform amounts of text data contributed by different attribute values (e.g., some authors may have written far more tweets than others but should not dominate in the representation).

Sometimes the number of tweets belonging to each attribute value does matter (e.g., if there are more tweets belonging to one hashtag than another, we might want to retain this difference). To accommodate such a need we further introduce *proportional topic weight* (PTW) which is defined as,

$$P(a_i|d) = \frac{|a_i|}{\sum_{j=1}^{M_d} |a_j|} \tag{5.9}$$

In PTW, we see that attributes with more text data would be weighted higher.

Finally, we may also have the *unweighted probability distributions* (UPD), defined as,

$$P(z = k|d) \propto \sum_{a_i \in d} P(z = k|a_i) \tag{5.10}$$

This weighting scheme encodes directly corpus-wide statistics, since there is no re-weighting of the attribute topic-document distributions.

Note that depending on the attribute a, it is possible that proportional, balance, and unweighted topic weights could be equal. The different pooling schemes for text document representations of the tweet messages we explored are as follows, a *single tweet message*, *pool tweets in a location*, *pool tweets by a single user*, and *pool tweets by hashtags*. Note that for the location we used both zip codes and counties. By providing topic models for these different attributes, we develop the multi-attribute topic features.

It is worth pointing out that the proposed multi-attribute topic features can also be constructed when there is no naturally available attribute, e.g. meta-data and for other count-based features. For example, if we are predicting the sentiment of news articles, then we can directly use the text for an article to generate features based on word counts, i.e., term frequency. Alternatively, we can also consider a different view of the articles, that is the news article is composed of many sections, which themselves contain paragraphs and those paragraphs contain many sentences. Thus, the prediction task can be decomposed to have

multi-view attributes, i.e., sections, paragraphs and sentences as well. While our methods are applied in the context of topic modeling, the approaches can also be used to amalgamate any numerical feature which is used for prediction, such as term-frequency counts.

5.5 EXPERIMENTS

For the remainder of this chapter, we evaluate the effectiveness of the proposed methods of constructing topic features by using a Twitter corpus of over 100 million tweets collected over a seven-year period in 2009-2015 to predict the new diagnosis rates of HIV, gonorrhea, and chlamydia at different temporal and spatial resolutions in the United States, at the zip code-level and county-level resolutions. The experimental results show that feature representations based on attributes such as authors, locations, and hashtags are generally more effective than the conventional topic feature representation without considering these multi-view attributes.

As the multi-view attribute features proposed are general for probabilistic distributions, they can be potentially used in any application of social media-based prediction to improve accuracy.

5.5.1 CDC STIs Corpus

In this section, we describe the data sets used for evaluating the proposed methods. The county-level HIV, chlamydia (CHLA), and gonorrhea (GONO) new diagnosis data are obtained from the Centers for Disease Control and Prevention (CDC) and AIDSVu [3]. Data are estimated for persons aged 13 and older living with an HIV infection diagnosis as of December 31st, of each respective year. Denominators used to calculate rates for county populations were obtained from the U.S. Census Bureau's census estimates for each respective year. Population denominators are restricted to persons aged 13 and older. Estimated rates of persons living with an HIV diagnosis were calculated per 100,000 population to permit data standardization and comparison. As is standard in the display of health statistics, rates generated from a numerator less than 12 are considered unstable and should be interpreted with caution. In the odd columns of Figure 5.5 we show the new diagnosis rates via each state in 2014, note the blank regions in the figure represents the suppressed data.

Philadelphia HIV New Diagnosis Dataset We obtained zip code-level HIV diagnosis rates per 100,000 from Philadelphia, Pennsylvania which the HIV data included only people

[3] http://aidsvu.org/

aged 13 and older. Data from regions with less than 5 new HIV diagnoses per year or less than 100 inhabitants are routinely suppressed by the CDC, and this suppression criteria were also applicable for the present analysis.

5.5.2 Twitter Data

Our Twitter corpus ranges from June 2009 to March 2010, November 2011 to December 2015. In total there were more than 3.4 billion tweets, including re-tweets. However, in order to use this dataset at the spatial granularity of the STI new diagnosis rates we geotagged our Twitter corpus to zipcodes, and counties, in the United States. The user geotagging problem has been well studied [183, 184]. In this study we developed a heuristic to quickly, and accurately, geotag tweets at the county and zip code resolutions.

Geo-location Tweets may contain geo-coordinates, e.g GPS, which we refer as coordinate data for short, and/or a "location" in the meta-data, we refer to location only data. We handle these two geotagging tasks separately, first we describe coordinate mapping and then location mapping: the mappings of those tweets without the coordinate information.

Coordinate Mapping The simplest tweets to geotag are those with coordinates data in tweets. Based on the zip code boundary shape, we first construct a minimal bounding rectangle (MBR) for each zip code and build hash tables storing the area the rectangle covers. Then for a point defined by the latitude and longitude pair, we find all possible zip codes and use ray casting algorithm to check which zip code contains the coordinates. This process can also be repeated for a different resolution such as US counties.

Location Only Mapping While it is impossible to geotag most users at an US zip code-level resolution, based on their provided location. We instead geotag these users at an US county-level resolution. We first applied a pre-processing procedure on the dataset, which included US time zone filtering and location field empty check. We then used a rule-based mapping, which mapped the location information of each tweet, based on some predefined rules. This approach is adapted from [75], in which select cities are mapped to counties if they contain at least 95% of the population of all the cities with the same name.

A more complete description of the geotagging method performance can be found in [86].

5.6 FORECASTING STIS PREVALENCE RATES USING TWITTER

The main purpose of our experiments was to examine two basic questions:

- Is the proposed multi-view attribute topic features more effective than the regular topic features (which are usually generated using one view)?
- Which of the proposed weighting functions performs the best?

These questions can be answered by comparing multiple runs with appropriate parameter configuration. As the baseline single view can be regarded as a special case of the proposed multi-view framework, the baseline method can be easily simulated by restricting to one view (e.g., pooling all tweets in a county in the case of the example illustrated in Figure 5.1), i.e. the natural document representation.

5.6.1 Data Pre-Processing

We selected three states from our CDC STI corpus which have higher level of STI new diagnosis rates compared to the rest of the country, i.e., these were California, Florida, and New York. We also included Pennsylvania for comparison with our Philadelphia analysis. We log-transformed and standardized these rates. Due to the quarterly nature of the Philadelphia HIV new diagnosis dataset, we included this time resolution for each attribute document representation of the Twitter data. We used a location-based representation, such as zip codes, then construct the four attribute documents, i.e., tweet messages are grouped by quarter belonging to the same zip code and corresponding to a HIV diagnosis rates.

This data also lends nicely to a semi-supervised framework, in which we used unlabeled text data to help guide the feature construction step. Specifically, we included all the Twitter data available for the state of Pennsylvania, regardless if we observed a new diagnosis, and made use of this unlabeled data within our supervised learning framework. We used topic modelling, which can be viewed as an unsupervised method for feature representation which clusters semantically similar documents, in our semi-supervised framework.

For all of our experiments we used LDA for topic modeling feature construction, normalized our multi-view attribute features and used an estimator, fitted on randomized decision trees (extra-trees) [185] for our regression problem.

To ensure there were no outliers in the Twitter dataset, we included the attribute documents, whose lengths (e.g., number of tweets) were within three standard deviations of the mean, and we used all the available new diagnosis testing data in order to compare the document representations. We only noticed the presence of outliers when considering the

Attribute Document	Weighting Schemes	HIV New Diagnosis			
		Florida	California	Pennsylvania	New York
Baseline	—	0.371	0.243	0.313	0.183
Quarterly	PTW	0.311	0.203	0.339	0.221
	BTW	0.399	0.238	0.262	0.277
	UPD	0.381	0.202	0.366	0.236
Authors	PTW	0.325	0.228	0.258	0.200
	BTW	0.300	0.176	0.248	0.126
	UPD	0.207*	0.137*	0.191	0.116*
Messages	PTW	0.296	0.172	0.292	0.154
	BTW	0.274	0.174	0.307	0.152
	UPD	0.228*	0.147*	0.180	0.114*
Hashtags	PTW	0.319	0.150*	0.245	0.170
	BTW	0.321	0.183	0.305	0.135
	UPD	0.248	0.145*	0.200	0.146
Train Size		228	168	135	150
Test Size		44	33	27	30

Table 5.1: Prediction MSEs for HIV new diagnosis, for four states with our proposed feature construction methods. A $*$ implies significant improvement with $\alpha = 0.1$, and $**$ is significant decrease with $\alpha = 0.1$ over the baseline.

authors, which follows a Zipfian distribution, i.e., a right skewed long tailed distribution and only excluded six authors which we manually verified were attributed to spam accounts.

5.6.2 CDC STIs Diagnosis County-level Prediction

We use the datasets prior to 2013 as training and considered the STI diagnosis for 2014 as the testing dataset. While the per-year STI diagnosis rates are only reported once a year, the tweets have a creation timestamp which allows us to pool messages by time, in we selected at a quarterly temporal resolution with all our attributes.

We propose a simple baseline, where all the messages pooled in a county for the entire year of 2014 is a document from which we constructed topics, which simulates a natural pooling strategy. This is a special case of our model where there is only a single attribute encompassing the entire document. We compared the topic features constructed using attributes with this baseline to see if multi-view topic features are indeed beneficial.

The training and testing sizes as well as the prediction mean-squared errors (MSE) are shown in Table 5.1, Table 5.2 and Table 5.3. We applied a two-sample t-test comparing the

Attribute Document	Weighting Schemes	Gonorrhea			
		Florida	California	Pennsylvania	New York
Baseline	—	0.461	0.300	1.023	1.033
Quarterly	PTW	0.511	0.432	1.015	1.150
	BTW	0.536	0.381	0.940	1.222
	UPD	0.592	0.318	0.941	1.333
Authors	PTW	0.413	0.443**	0.822	0.692*
	BTW	0.377	0.283	0.882	0.733
	UPD	0.354	0.325	0.736*	0.610*
Messages	PTW	0.379	0.414	0.794	0.620*
	BTW	0.364	0.392	0.819	0.624*
	UPD	0.408	0.471**	0.638*	0.584*
Hashtags	PTW	0.403	0.480**	0.841	0.690*
	BTW	0.377	0.561**	0.854	0.581*
	UPD	0.365	0.515**	0.838	0.658*
Train Size		304	234	256	260
Test Size		64	57	63	60

Table 5.2: Prediction MSEs for Gonorrhea, for four states with our proposed feature construction methods. A $*$ implies significant improvement with $\alpha = 0.1$, and $**$ is significant decrease with $\alpha = 0.1$ over the baseline.

attribute document and weighting scheme result with the baseline and noted results with significant improvement over the baseline or significant decrease in performance compared to the baseline.

We observe that UPD obtains the minimum MSE, which is not too surprising since the diagnosis rates tend to be concentrated in the metropolitan areas as shown in Figure 5.5, and UPD was constructed to favor populous locations. We also see that the BTW under the author attribute always improves over the baseline. Partitioning by time helps when the training dataset is small, even though HIV new diagnosis for the states is the sparsest of all STI new diagnosis, we can still achieve good performance with the Quarterly attribute document. Gonorrhea new diagnosis rates are the most difficult to predict, especially in California which only by using authors and the BTW scheme can we outperform the baseline. Overall using the attributes message and authors yield the best results in particular authors in Florida and California, which have a non-uniform STI-rates distribution and messages were best for Pennsylvania and New York which tend to be more mostly uniform, with few peaks.

Attribute Document	Weighting Schemes	Chlamydia			
		Florida	California	Pennsylvania	New York
Baseline	—	0.144	0.141	0.259	0.150
Quarterly	PTW	0.178	0.119	0.232	0.168
	BTW	0.190	0.146	0.205	0.182
	UPD	0.203**	0.100	0.199	0.183
Authors	PTW	0.146	0.110	0.165*	0.100*
	BTW	0.143	0.104	0.196	0.088*
	UPD	0.107	0.103	0.155*	0.086*
Messages	PTW	0.129	0.093	0.134*	0.082*
	BTW	0.129	0.100	0.167*	0.077*
	UPD	0.140	0.073*	0.163*	0.095*
Hashtags	PTW	0.124	0.087*	0.160*	0.106
	BTW	0.143	0.107	0.167*	0.101*
	UPD	0.137	0.104	0.155*	0.087*
Train Size		308	267	320	297
Test Size		64	58	67	61

Table 5.3: Prediction MSEs for Chlamydia, for four states with our proposed feature construction methods. A $*$ implies significant improvement with $\alpha = 0.1$, and $**$ is significant decrease with $\alpha = 0.1$ over the baseline.

5.6.3 Philadelphia Zipcode-level Prediction

Using the available data prior to 2015 (2009-2014) as our training dataset and for the testing data we choose the most recent HIV new diagnosis data in 2015. We tuned our parameters on a development set, which included the Philadelphia zip code 2014 HIV new diagnosis data for evaluation and the data prior as the training dataset. The training data contained 352 entries, of which 156 were non-missing, and the test data contained 74 entries of which 44 were non-missing.

We used both the mean squared error (MSE) and median squared error as our error metrics for the Philadelphia prediction. We compare our weighting scheme in Table 5.4, by predicting the HIV new diagnosis rates directly for each zip code. A clear pattern from these results is that the UPD performed the worst in almost all cases. The UPD scheme distributes the topic weights to the populous locations and thus relying on having enough tweet messages to represent this distribution.

While both PTW and BTW outperform UPD, both schemes are similar in performance. But when considering authors as attribute documents, BTW has an overall better MSE score than the other schemes. Such results indicate that partitioning by authors works consistently

Figure 5.5: Top to bottom: Florida, California, Pennsylvania, New York. Left most two columns: HIV New diagnosis, Middle: Gonorrhea, Right: Chlamydia, predictions for 2014 incident rates, via Authors and UDP scheme.

Attribute Document	Weighting Schemes	Errors	
		mean SE	median SE
Zipcodes	PTW/ BTW/ UPD	18.32	6.01
Author	PTW	15.87	10.24
	BTW	14.93	9.80
	UPD	18.07	10.40
Hashtag	PTW	19.68	14.75
	BTW	19.86	14.75
	UPD	22.77	16.00
Message	PTW	16.64	9.42
	BTW	16.63	9.67
	UPD	17.81	8.21

Table 5.4: Overall HIV new diagnosis prediction results by weighting scheme

well, since it avoids the bias from dominance by authors who wrote many more tweets than others (i.e., less biased due to variable data size).

5.6.4 Topic Features Population Bias

We have previously alluded to the population-bias as the effect of depending on message count statistics to produce useful features. We measured this population bias for the CDC STIs county-level prediction by computing the Pearson correlation coefficient with respect to each topic feature and the county tweet message counts. We plot the absolute correlation lower bound and the percentage of features which have a correlation coefficient, whose absolute value is greater than the lower bound for the author attribute and for the GONO testing dataset in Figure 5.6, e.g., at lower bound of 0; all the topic features are shown, and no feature has above a correlation coefficient of 1. Although not shown the other attributes follow a similar pattern.

We observe that our UDP indeed creates features which are population biased, having a strong message count correlation with more than 90 of all the features. It is also interesting to note that the baseline has about 40 features with a weak correlation (0.2-0.4) for all states except California. Both BTW and PTW do not show this type of association and tend to plateau at 0 before the baseline. We find a similar association with the zip code features as well. Thus, depending on the prediction problem constructing predictive features, UPD could be useful, however if we are interested in making a more robust feature, invariant to the number of messages in some attribute, then it may be better to use the BTW scheme while sacrificing some prediction accuracy.

Attribute Feature Comparison While the author attribute features tend to work better with smaller training sample sizes, using messages attribute features in general will work well. It is somewhat surprising that hashtags do not perform quite on par as authors since, when pooling by hashtags we can expect to create coherent documents. One explanation could stem from the fact that there are many infrequent, as well as very popular hashtags thus causing some disparity in the document sizes. Another factor could be that hashtags are more susceptible to the language shift, since there could be many new events specific to 2015. Thus, to measure the topic cohesion we compute the log perplexity of the attributes.

	Quarterly	Message	Author	Hashtags
Log Perplexity	231.89	20.74,	25.07	22.50

Table 5.5: Log Perplexity for different document attributes.

Table 5.5 presents the log perplexity, so called per-word likelihood bound, for all four different document attributes, Counties, Messages, Hashtags and Authors. To compute the perplexity, we used a withheld development dataset consisting of location only mapped tweets, meaning could not map to zip codes, instead we used county-based mappings for each quarter in 2015. to construct the document attributes. The perplexity for the quarterly attribute is much worse that the rest, which could be expected since pooling based on time may not necessarily create the most coherent documents. While Hashtags and Messages fit better the development data, it doesn't mean that this is able to translate to the predictive accuracy.

Hashtag Attribute Feature Analysis As a qualitative study we show the hashtags attribute topic features in word clouds, see Figure 5.7 in order to better observe the topic clusters. We used the topic predictor weights, obtained from our learning algorithm, and selected the top-2 weighted topics, based on the Philadelphia dataset, we then ranked the hashtags themselves based on their weights for these topics and selected the top 20 STI-related hashtags in Figure 5.7. To identify the STI-related hashtags we used a manually curated STI-related terms to filter hashtags which contain these terms. The hashtags in Figure 5.7 are all within the top 10% highest ranked hashtags. We find that many indeed are related to sexual themes, e.g., #casualsexweek, but further study is needed to understand in what context and if it is indicative of risky behavior.

5.7 RELATED WORK

In this section we review relevant work for both multi-view attribute features and health related prediction tasks.

5.7.1 Multi-view Learning

Multi-view learning first introduced in the semi-supervised setting by [186] and [187]. Yarowsky in [187], described an unsupervised word disambiguation algorithm which takes two views (senses) of words, one view is the context of the word (collocation) and is given by one-sense-per-discourse view. Blum and Mitchell in [186] formalize the notion of multi-view learning in the context of web-page classification. They take two views of webpages, the anchor text and the content of the webpages to develop two learning algorithms from each view. They use the output of one classifier to enhance the training data of the other, this method

is called co-training. The multi-view learning has also been extended for unsupervised data, specifically to clustering using mixture models [188].

The idea of multi-view learning has been used in the vision community as well, where the goal is to represent different feature types in some unified framework [189, 190, 191]. For example combining BOW features with embedding features, it would not make sense to concatenate these two feature types as the dimensions in these features represent something completely different, that is they have different statistical properties. Instead in they can be first mapped to some common low-dimensional subspace.

The notion of multi-view learning is orthogonal to multi-view attribute features, as the goal for multi-view attribute features is to create a document feature representation from the consensus of the attribute partitions. In other words, we focus on maximizing the benefit of a single feature type while multi-view learning deals with multiple feature types and the best way to combine them.

5.7.2 Health-Related Prediction Tasks

To the best of our knowledge, no previous work has studied how to use meta data to construct multi-view topic features for social media health-based prediction. The closest work to ours is the use of topic modeling for tweets in prediction tasks. In this line, topic modeling has also been employed, with some success, in predicting heart disease mortality at the county-level using Twitter [192] and to analyze the language and personality traits on Facebook [87]. In both works the authors applied topic modelling directly on to discover topics from Twitter and Facebook messages. They then used these 2,000 Twitter topics to estimate a user-level [87], and county-level [192] topic weights by weighing every word in a message, by the topic proportion and topic distributions. Although they claimed to discover high quality topics, other studies yielded low quality topics using such approach [179]. Our work proposes a general framework and multiple new strategies for topic feature construction that are shown to perform better than these ad hoc topic feature construction methods.

Twitter as a useful social media information source has been proven adequate for many health-related tasks such as the prediction of suicide [193], influenza rates [194, 195, 196], asthma-related emergency room visits [197], and HIV rates [73, 74, 75, 181]. However, there are only a few works using tweets to predict public health issues [73, 193, 198, 199, 200, 201]. Few works have used topic modeling approaches for predicting health-related outcomes [86, 87, 192].

Some studies use specific keywords such as the words “flu”, “influenza”, and associated

symptoms like "high fever" [196, 202] to predict flu and influenza trends. While others have used dictionary-based approaches for HIV prevalence rate prediction [73, 74, 75]. For example, in [73] the authors used two dictionaries related to sexual risk behaviors and attitudes; they classified tweets being drug related or sex related messages, if they contained at least one corresponding risk-related term and finally they used the number of risk-related related tweets as an input feature for a down-stream regression task. In our work, we applied different text mining strategies to construct useful features that go beyond term count. We made use of the semantic structure in tweets and built topic models which can be aligned to locations and showed how we can develop features, for predicting HIV and other STIs, which are not limited to a closed-vocabulary approach.

Further, some have proposed different schemes for training the development of new models to improve the topics quality. [55, 178, 179, 203, 204]. Hong and Davison [179] used different aggregation strategies to overcome the short message limitation. They show that the induced topic models are a good feature for classification problems. Alvarez-Melis and Saveski [178] compared of different pooling methods, including at the user, hashtag, and conversations level. They show that more coherent topics and also helped in document retrieval tasks, however it also hurt running time performance for creating topic models. In brief, our work proposes a more general framework that include multiple new strategies for topic feature construction by exploiting meta-data attributes, and furthermore we examine the performance of the features for health-based prediction tasks.

5.8 SUMMARY

In this chapter, we address a fundamental problem in all those prediction applications, i.e., how to construct effective model-based features in the presence of the outcome misalignment problem and proposed a novel framework for constructing multi-view topic features by leveraging a topic model as a building block. The multi-view topic features are constructed based on the multiple attributes of social media data that are naturally available and can be regarded as attribute features. We propose and study three different weighting scheme methods for our multi-view attribute features, i.e., unweighted, balanced and proportional, each make different underlying assumptions about how the data is distributed and act as regularization methods. We evaluated the proposed methods using an application on the public health domain prediction of STIs using tweets, and showed pooling by attributes, such as authors, outperformed the baseline in prediction. The results show that attribute-based multi-view topic features are consistently more effective than the baseline single-view features. Although the framework is proposed for social media-based prediction, it is general

in that the attributes can be defined based on any meta-data available in text-based prediction applications. As the proposed framework is general, another very interesting direction for future work is to explore the application of the general framework in other social media domains.

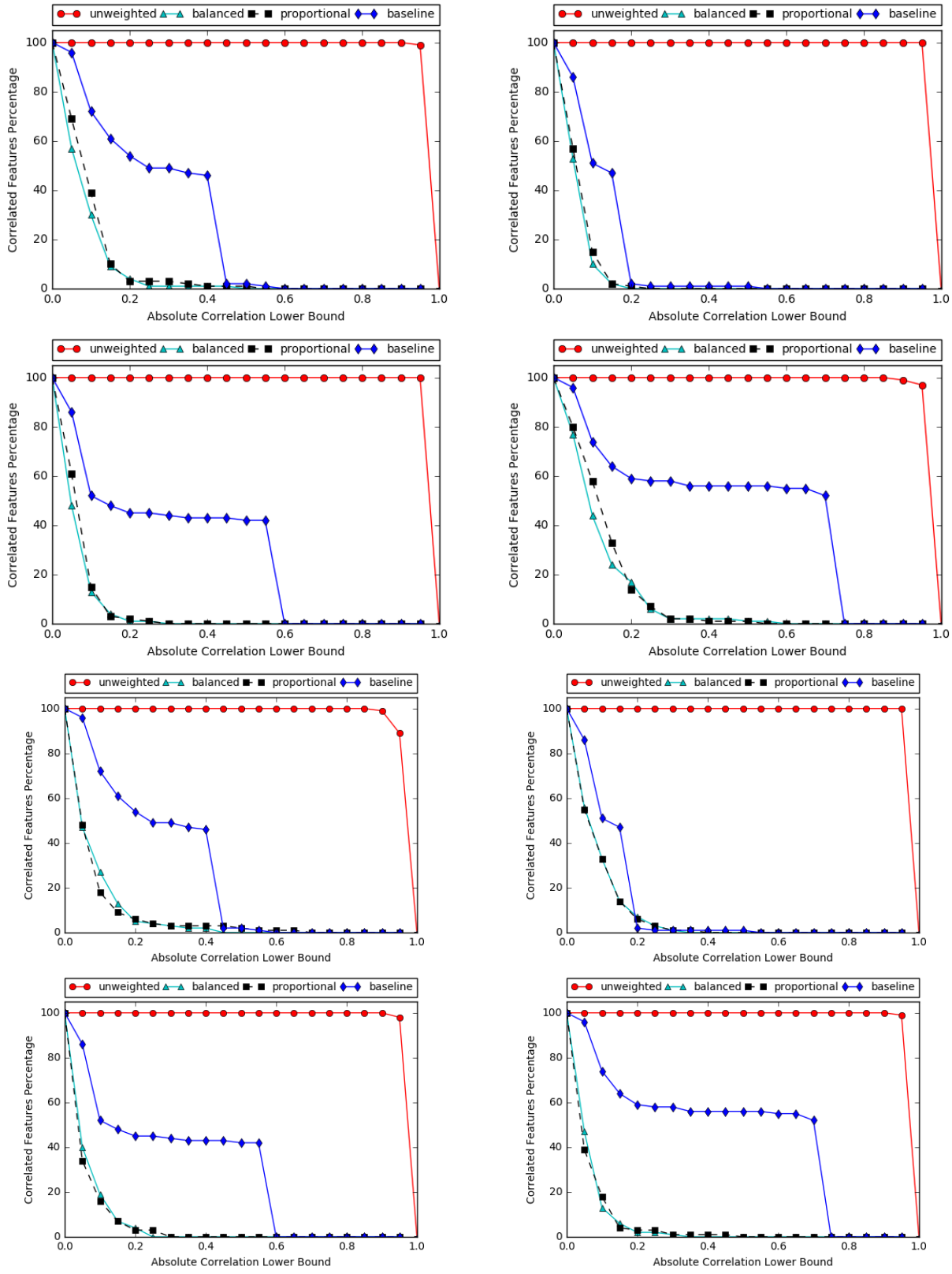


Figure 5.6: Feature-Message correlations for FL, CA, PA, and NY (left to right), using the Author, and Hashtags (top to bottom) attributes.

Figure 5.7: The highest weighted, top topics for Philadelphia zipcodes, with the top-20 highest weighted hashtags, using the UDP scheme.

CHAPTER 6: CONCLUSION AND FUTURE WORK

We explored various lines of work in model-based feature construction. However, we are still far from generating a comprehensive method for model-based feature construction and there is some groundwork left to be done to reach our goal of automatic model-based feature construction.

6.1 SUMMARY OF OVERALL CONTRIBUTIONS

In this dissertation we described model-based feature construction in social media data, the contributions can be best summarized by Figure 2.2. In each line of work, we used social media text along with some associated data context to construct model-based features. In Chapter 3, we proposed a mixture model to compute differential semantic features in the context of humorous text identification. The probabilistic model was used to compare a source text to a reference corpus and generated a distribution over the reference entities. We showed that we can use these features to generate incongruity and unexpectedness features for humor identification. In Chapter 4 we used user comments and posting patterns to model source reliabilities. This unsupervised model was able to learn trustworthy comment embeddings for every question, fine-tuned word embeddings from the comments and learned fine-grained user aspect reliabilities. We showed that these features can also be used for other tasks such as expert classification of users, as they outperform topic features. In Chapter 5 we proposed a framework that leverages meta-data information in social media to construct multi-view attribute features of the text data. In doing so we developed a solution for the target misalignment problem on social media for STI new diagnosis prediction. This framework can also leverage topic-based features to generate new features based on multi-views of the social media data.

We have investigated these models in different social media domains. In the humor identification task, we consider humorous reviews from Yelp and connected this source text with Wikipedia as a reference corpus. In identifying trustworthy comments, we leveraged Reddit subreddit communities to construct the CrowdQM AskReddit Dataset, which included data from AskDocs, AskHistorians and AskScience. Finally, in the multi-view attribute features we leveraged data from Twitter and predicted new diagnosis from various STIs including HIV, Gonorrhea and Chlamydia.

6.1.1 Limitations

Our works have demonstrated some success in leveraging different context of social media data for model-based feature construction, however there is little guidance on how exactly to create a model for a new context. In our case we have shown that it is possible to create models with a graphical mixture models as well as an optimization framework. We also proposed a multi-view framework to allow the reuse of probabilistic model-based features in the presence of meta-data on social media. In general, using text derived features for prediction tasks tries to associate derived features from the text to the outcome variable. In doing so, we are asking the question "what variables is most associated with the outcome?", however this only reflects what our feature input representation can capture. In other words, once we have discovered the most discriminative features, we cannot conclude that the feature is indicative of the outcome variable, in order to do so we must analyze the causal reasons for that. This is also true for model-based features that can encode different contexts of the data, however, as opposed to deep learning or neural network features, the features derived from MBFC, if the underlying model is a graphical model, can include causal relationships.

6.2 FUTURE DIRECTIONS

In this dissertation we showed how we can use model-based feature construction for prediction tasks in social media data. The potential for model-based features to go beyond text prediction is what makes MBFC particularly appealing for other domains and tasks. However, before we can apply these methods to other tasks there are some limitations which we would need to address particularly in interpretability and integration of neural network approaches to MBFC.

6.2.1 Model Interpretability

The potential for interpretable features is what makes model-based features potentially more attractive than deep learning features. While we have showed some benefit of the features generated, we have not thoroughly evaluated the features for interpretability. While topic features can generate useful clusters, they might be less semantically meaningful for humans [205]. Thus, to further measure the interpretability for the model-based features, it is also important to collect human judgement on the model outputs and devise better evaluation metrics for semantic meanings of word clusters.

6.2.2 Towards Automatic Model-Based Feature Construction for Embeddings

In chapter 5, we introduced multi-view aspect features, the objective was:

- To leverage an existing model, i.e., topic modeling and construct appropriate features for social media when there is a target misalignment problem.
- To develop a probabilistic multi-view representation of the data, to facilitate model-based feature construction in social media.

One main observation we found was that we can identify the best perspective to apply topic modeling by considering the coherence of the view, i.e., measuring how well the grouping of text data makes sense. While the focus for the STI prediction problem was topic modeling, the multi-view approach to model-based feature construction is general and can also be applied to other models for feature development.

A natural question is how to we leverage research in neural network and deep learning to domains to extend the multi-view approach. One direction of research is to apply this approach to word embedding features. However, this raises new challenges, not present in the multi-view aspect feature construction method such as how to weigh the importance of word, or character-level, units for which we have feature representations. Thus, the probabilistic representation may be limited, since it does not capture the real-valued representation for embedding. A recent work, [206], has shown it is possible to use multi-views for sbstractive dialogue summarization, however it is not clear how to adapt this to a prediction framework on social media.

In particular we can consider the following questions for future work:

- What is the best way to aggregate the embedding-based features? There are several ways to combine the embedding-based features, for example if we take a direct implementation of our probabilistic multi-view approach, then for each "document" partition we could perform a weighted aggregation.
- Can we use the same coherence measure in order to give us a good view of the data to construct our embedding-based features? We would need to develop new metrics to use if the coherence is no longer valid.

However an alternative approach is to develop a deep learning architecture to represent each intermediate partition [96]. The bottleneck here is then developing a unique deep learning architecture for each of the representations.

6.2.3 Studying User-User interactions

From previous work, we saw that we can learn useful signals by incorporating background text to ground the text to some background knowledge. We can also leverage some structure in community question answering forums to learn expertise of users. However, to learn user-reliabilities, one limitation is that there is an assumed structure in social media, i.e., someone asks a question and users give appropriate answers to try to answer this question. However, in many cases this may not be given explicitly, for instance in Twitter there may be many users talking about vaccine hesitancy, i.e., antivax supporters, and while some may raise legitimate concerns, there are many propagating false information. Thus, an extension of source reliability modeling in social media is to model reliabilities directly from user-user interactions.

APPENDIX A: CROWDQM SUPPLEMENTARY MATERIAL

A.1 DERIVATION OF THE CROWDQM UPDATE RULES

Recall we use coordinate descent [160] to solve our optimization problem, hence we split the update rules to three cases.

Case A.1. We can rewrite the objective function in Equation 4.5 as follows

$$\begin{aligned}&\min_{a_m^*} \sum_{m=1}^{M} \sum_{n \in \mathcal{N}_m} R_{m,n} ||a_m^* - a_{m,n}||^2 + C \\ &\textbf{s.t. } R_{m,n} = \sum_{k=1}^{K} r_n^{(k)} d(u_n^{(k)}, p_m^{(k)}) \textbf{ and } a_{m,n} = \frac{1}{|w_{m,n}|} \sum_{\omega \in w_{m,n}} v_\omega \textbf{ and } \sum_n \exp(-r_n^{(k)}) = 1 \forall k\end{aligned} \tag{A.1}$$

where C is a constant w.r.t. a_m^*, taking the derivative of Equation A.1 and setting it equal to zero we have

$$\frac{\partial f}{\partial a_m^*} = \sum_{n \in \mathcal{N}_m} 2R_{m,n} \cdot (a_m^* - a_{m,n}) = 0 \tag{A.2}$$

and therefore,

$$a_m^* = \frac{\sum_{n \in \mathcal{N}_m} R_{m,n} a_{m,n}}{\sum_{n \in \mathcal{N}_m} R_{m,n}} \tag{A.3}$$

Case A.2. Re-writing the objective function in Equation 4.5, we have

$$\begin{aligned}\min_{r_n^{(k)}} \sum_{n=1}^{N} R_{m,n} \left(||a_m^* - a_{m,n}||^2 + \frac{\beta}{|c_m|} \sum_{c \in c_m} ||a_{m,n} - v_c||^2 \right) \\ \textbf{s.t. } \sum_n \exp(-r_n^{(k)}) = 1 \forall k\end{aligned} \tag{A.4}$$

Taking the derivative of the Lagrangian of Equation A.4 and setting it to zero we have

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial r_n^{(k)}} = &\sum_{m \in \mathcal{M}_n} d(u_n^{(k)}, p_m^{(k)}) \left(||a_m^* - a_{m,n}||^2 + \frac{\beta}{|c_m|} \sum_{c \in c_m} ||a_{m,n} - u_c||^2 \right) \\ &- \lambda_k \exp(-r_n^{(k)}) = 0\end{aligned} \tag{A.5}$$

Hence we can re-arrange Equation A.5 solving for $r_n^{(k)}$,

$$r_n^{(k)} = -\log \sum_{m \in \mathcal{M}_n} d(u_n^{(k)}, p_m^{(k)}) \left(||a_m^* - a_{m,n}|| + \frac{\beta}{|c_m|} \sum_{c \in c_m} ||a_{m,n} - v_c||^2 \right) + \log \lambda_k \tag{A.6}$$

the last term is a normalization term for each k.

Case A.3. We rewrite Equation 4.5 summing over the post-comment pairs

$$\min_{v_\omega} \sum_{<m,n> \in D_\omega} R_{n,m} \left(||a_m^* - a_{m,n}||^2 + \frac{\beta}{|c_m|} \sum_{c \in c_m} ||a_{m,n} - v_c||^2 \right). \tag{A.7}$$

Taking the derivative of Equation A.7 with respect to v_ω and setting the resulting equation equal to zero we have

$$\sum_{<m,n> \in \mathcal{D}_\omega} \frac{R_{m,n}}{|w_{m,n}|} \left(a_m^* + \frac{\beta}{|c_m|} \sum_{c \in c_m} v_c \right) - \frac{R_{m,n}(1+\beta)}{|w_{m,n}|^2} \sum_{\omega' \in w_{m,n}} v_{\omega'} = 0 \tag{A.8}$$

Note that,

$$\sum_{\omega' \in w_{m,n} s} v_{\omega'} = v_\omega + \sum_{\omega' \in w_{m,n} s \setminus \{\omega\}} v_{\omega'}. \tag{A.9}$$

Thus we can rewrite Equation A.8 as

$$\begin{aligned} \sum_{<m,n> \in \mathcal{D}_\omega} \frac{R_{m,n}(1+\beta)}{|w_{m,n}|^2} v_\omega = & \sum_{<m,n> \in \mathcal{D}_\omega} \frac{R_{m,n}}{|w_{m,n}|} \left(a_m^* + \frac{\beta}{|c_m|} \sum_{c \in c_m} v_c \right) \\ & - \frac{R_{m,n}(1+\beta)}{|w_{m,n}|^2} \sum_{\omega' \in w_{m,n} \setminus \{\omega\}} v_{\omega'} \end{aligned} \tag{A.10}$$

solving for v_ω then follows directly.

REFERENCES

[1] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "A survey on deep learning for big data," *Information Fusion*, vol. 42, pp. 146–157, 2018.

[2] F. Carcillo, Y.-A. Le Borgne, O. Caelen, Y. Kessaci, F. Oblé, and G. Bontempi, "Combining unsupervised and supervised learning in credit card fraud detection," *Information Sciences*, 2019.

[3] J. Cao, X. Jiang, B. Zhao et al., "Mathematical modeling and epidemic prediction of covid-19 and its significance to epidemic prevention and control measures," *Journal of Biomedical Research & Innovation*, vol. 1, no. 1, pp. 1–19, 2020.

[4] S. Yousefinaghani, R. Dara, Z. Poljak, T. M. Bernardo, and S. Sharif, "the assessment of twitter's potential for outbreak detection: Avian influenza case study," *Scientific reports*, vol. 9, no. 1, pp. 1–17, 2019.

[5] E. A. Evans, E. Delorme, K. Cyr, and D. M. Goldstein, "A qualitative study of big data and the opioid epidemic: recommendations for data governance," *BMC Medical Ethics*, vol. 21, no. 1, pp. 1–13, 2020.

[6] C. Zhang, P. Patras, and H. Haddadi, "Deep learning in mobile and wireless networking: A survey," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2224–2287, 2019.

[7] C. M. Bishop, *Pattern recognition and machine learning*. Springer Science+ Business Media, 2006.

[8] M. Färber and A. Jatowt, "Citation recommendation: Approaches and datasets," *arXiv preprint arXiv:2002.06961*, 2020.

[9] J. Heaton, "An empirical analysis of feature engineering for predictive modeling," in *SoutheastCon 2016*. IEEE, 2016, pp. 1–6.

[10] D. D. Lewis, "Representation and learning in information retrieval," 1992.

[11] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal of research and development*, vol. 2, no. 2, pp. 159–165, 1958.

[12] J. L. Fagan, "Experiments in automatic phrase indexing for document retrieval: a comparison of syntactic and non-syntactic methods," Cornell University, Tech. Rep., 1987.

[13] D. A. Evans and C. Zhai, "Noun-phrase analysis in unrestricted text for information retrieval," in *Proceedings of the 34th annual meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 1996, pp. 17–24.

[14] E. Stamatatos, "A survey of modern authorship attribution methods," *Journal of the American Society for information Science and Technology*, vol. 60, no. 3, pp. 538–556, 2009.

[15] 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, 1990.

[16] T. Hofmann, "Probabilistic latent semantic indexing," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 50–57.

[17] Y. Lv and C. Zhai, "Adaptive term frequency normalization for bm25," in *Proceedings of the 20th ACM international conference on Information and knowledge management*, 2011, pp. 1985–1988.

[18] Y. Lv and C. Zhai, "When documents are very long, bm25 fails!" in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, 2011, pp. 1103–1104.

[19] H. K. Kim and M. Kim, "Model-induced term-weighting schemes for text classification," *Applied Intelligence*, vol. 45, no. 1, pp. 30–43, 2016.

[20] A. Chowdhury and M. C. McCabe, "Improving information retrieval systems using part of speech tagging," Tech. Rep., 1998.

[21] R. Mandala, T. Tokunaga, and H. Tanaka, "The use of wordnet in information retrieval," in *Usage of WordNet in Natural Language Processing Systems*, 1998.

[22] Y. Shibata, T. Kida, S. Fukamachi, M. Takeda, A. Shinohara, T. Shinohara, and S. Arikawa, "Byte pair encoding: A text compression scheme that accelerates pattern matching," Technical Report DOI-TR-161, Department of Informatics, Kyushu University, Tech. Rep., 1999.

[23] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," *arXiv preprint arXiv:1508.07909*, 2015.

[24] M. Schuster and K. Nakajima, "Japanese and korean voice search," in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 5149–5152.

[25] T. Kudo, "Subword regularization: Improving neural network translation models with multiple subword candidates," *arXiv preprint arXiv:1804.10959*, 2018.

[26] A. M. Robertson and P. Willett, "Applications of n-grams in textual information systems," *Journal of Documentation*, vol. 54, no. 1, pp. 48–67, 1998.

[27] A. Järvelin, A. Järvelin, and K. Järvelin, "s-grams: Defining generalized n-grams for information retrieval," *Information Processing & Management*, vol. 43, no. 4, pp. 1005–1019, 2007.

[28] U. Sapkota, S. Bethard, M. Montes, and T. Solorio, "Not all character n-grams are created equal: A study in authorship attribution," in *Proceedings of the 2015 conference of the North American chapter of the association for computational linguistics: Human language technologies*, 2015, pp. 93–102.

[29] Y. Mehdad and J. Tetreault, "Do characters abuse more than words?" in *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 2016, pp. 299–303.

[30] Z. Zhang and L. Luo, "Hate speech detection: A solved problem? the challenging case of long tail on twitter," *Semantic Web*, vol. 10, no. 5, pp. 925–945, 2019.

[31] H. E. Boukkouri, O. Ferret, T. Lavergne, H. Noji, P. Zweigenbaum, and J. Tsujii, "Characterbert: Reconciling elmo and bert for word-level open-vocabulary representations from characters," *arXiv preprint arXiv:2010.10392*, 2020.

[32] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[33] A. Banerjee and S. Basu, "Topic models over text streams: A study of batch and online unsupervised learning," in *Proceedings of the 2007 SIAM International Conference on Data Mining*. SIAM, 2007, pp. 431–436.

[34] S. P. Crain, K. Zhou, S.-H. Yang, and H. Zha, "Dimensionality reduction and topic modeling: From latent semantic indexing to latent dirichlet allocation and beyond," in *Mining text data*. Springer, 2012, pp. 129–161.

[35] R. Alghamdi and K. Alfalqi, "A survey of topic modeling in text mining," *Int. J. Adv. Comput. Sci. Appl.(IJACSA)*, vol. 6, no. 1, 2015.

[36] H. Jelodar, Y. Wang, C. Yuan, X. Feng, X. Jiang, Y. Li, and L. Zhao, "Latent dirichlet allocation (lda) and topic modeling: models, applications, a survey," *Multimedia Tools and Applications*, vol. 78, no. 11, pp. 15 169–15 211, 2019.

[37] 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 empirical methods in natural language processing*, 2009, pp. 248–256.

[38] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[39] X. Rong, "word2vec parameter learning explained," *arXiv preprint arXiv:1411.2738*, 2014.

[40] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[41] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[42] S. Serrano and N. A. Smith, "Is attention interpretable?" *arXiv preprint arXiv:1906.03731*, 2019.

[43] A. Jacovi and Y. Goldberg, "Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness?" in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, July 2020, pp. 4198–4205.

[44] A. Nijholt, "Towards humor modelling and facilitation in smart environments," *Advances in Affective and Pleasurable Design*, pp. 260–269, 2014.

[45] Y. Lu, H. Wang, C. Zhai, and D. Roth, "Unsupervised discovery of opposing opinion networks from forum discussions," in *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2012, pp. 1642–1646.

[46] X. Liu, J. Liu, and H. Chen, "Identifying adverse drug events from health social media: a case study on heart disease discussion forums," in *International conference on smart health*. Springer, 2014, pp. 25–36.

[47] M. Luca, "Reviews, reputation, and revenue: The case of yelp. com," *Com (March 15, 2016). Harvard Business School NOM Unit Working Paper*, no. 12-016, 2016.

[48] Y. Xiong, M. Cho, and B. Boatwright, "Hashtag activism and message frames among social movement organizations: Semantic network analysis and thematic analysis of twitter during the# metoo movement," *Public Relations Review*, vol. 45, no. 1, pp. 10–23, 2019.

[49] P. Candon, "Twitter: Social communication in the twitter era," 2019.

[50] T. G. Massoud, J. A. Doces, and C. Magee, "Protests and the arab spring: An empirical investigation," *Polity*, vol. 51, no. 3, pp. 000–000, 2019.

[51] D. A. McDonald, "Framing the "arab spring": Hip hop, social media, and the american news media," *Journal of Folklore Research*, vol. 56, no. 1, pp. 105–130, 2019.

[52] J. L. Blevins, J. J. Lee, E. E. McCabe, and E. Edgerton, "Tweeting for social justice in# ferguson: Affective discourse in twitter hashtags," *New Media & Society*, p. 1461444819827030, 2019.

[53] G. Li, J. Wang, Y. Zheng, and M. Franklin, "Crowdsourced data management: A survey," in *Data Engineering (ICDE), 2017 IEEE 33rd International Conference on*. IEEE, 2017, pp. 39–40.

[54] A. Morales, K. Narang, H. Sundaram, and C. Zhai, "Crowdqm: Learning aspect-level user reliability and comment trustworthiness in discussion forums," in *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer, 2020, pp. 592–605.

[55] A. Morales and C. Zhai, "Identifying humor in reviews using background text sources," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2017, pp. 503–512.

[56] A. Morales, N. Gandhi, M. S. Chan, S. Lohmann, T. Sanchez, K. A. Brady, L. Ungar, D. Albarracín, and C. Zhai, "Multi-attribute topic feature construction for social media-based prediction," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018, pp. 1073–1078.

[57] M. Ott, Y. Choi, C. Cardie, and J. T. Hancock, "Finding deceptive opinion spam by any stretch of the imagination," in *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1*. Association for Computational Linguistics, 2011, pp. 309–319.

[58] M. M. Mirończuk and J. Protasiewicz, "A recent overview of the state-of-the-art elements of text classification," *Expert Systems with Applications*, vol. 106, pp. 36–54, 2018.

[59] M. Haddoud, A. Mokhtari, T. Lecroq, and S. Abdeddaïm, "Combining supervised term-weighting metrics for svm text classification with extended term representation," *Knowledge and Information Systems*, vol. 49, no. 3, pp. 909–931, 2016.

[60] Z. Wu, H. Zhu, G. Li, Z. Cui, H. Huang, J. Li, E. Chen, and G. Xu, "An efficient wikipedia semantic matching approach to text document classification," *Information Sciences*, vol. 393, pp. 15–28, 2017.

[61] A. Onan, S. Korukoğlu, and H. Bulut, "Ensemble of keyword extraction methods and classifiers in text classification," *Expert Systems with Applications*, vol. 57, pp. 232–247, 2016.

[62] A. S. Manek, P. D. Shenoy, M. C. Mohan, and K. Venugopal, "Aspect term extraction for sentiment analysis in large movie reviews using gini index feature selection method and svm classifier," *World wide web*, vol. 20, no. 2, pp. 135–154, 2017.

[63] A. Bandhakavi, N. Wiratunga, D. Padmanabhan, and S. Massie, "Lexicon based feature extraction for emotion text classification," *Pattern recognition letters*, vol. 93, pp. 133–142, 2017.

[64] C. D. Manning, H. Schütze, and P. Raghavan, *Introduction to information retrieval*. Cambridge university press, 2008.

[65] G. Salton, A. Wong, and C.-S. Yang, "A vector space model for automatic indexing," *Communications of the ACM*, vol. 18, no. 11, pp. 613–620, 1975.

[66] A. I. Kadhim, "Term weighting for feature extraction on twitter: A comparison between bm25 and tf-idf," in *2019 International Conference on Advanced Science and Engineering (ICOASE)*. IEEE, 2019, pp. 124–128.

[67] Y. Champclaux, T. Dkaki, and J. Mothe, "Enhancing high precision by combining okapi bm25 with structural similarity in an information retrieval system." in *ICEIS (3)*, 2009, pp. 279–285.

[68] B. Aklouche, I. Bounhas, and Y. Slimani, "Bm25 beyond query-document similarity," in *International Symposium on String Processing and Information Retrieval*. Springer, 2019, pp. 65–79.

[69] T. Sabbah, A. Selamat, M. H. Selamat, F. S. Al-Anzi, E. H. Viedma, O. Krejcar, and H. Fujita, "Modified frequency-based term weighting schemes for text classification," *Applied Soft Computing*, vol. 58, pp. 193–206, 2017.

[70] K. Chen, Z. Zhang, J. Long, and H. Zhang, "Turning from tf-idf to tf-igm for term weighting in text classification," *Expert Systems with Applications*, vol. 66, pp. 245–260, 2016.

[71] Y. Wang and H. Y. Youn, "Feature weighting based on inter-category and intra-category strength for twitter sentiment analysis," *Applied Sciences*, vol. 9, no. 1, p. 92, 2019.

[72] M. Lan, C. L. Tan, J. Su, and Y. Lu, "Supervised and traditional term weighting methods for automatic text categorization," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, no. 4, pp. 721–735, 2008.

[73] S. D. Young, C. Rivers, and B. Lewis, "Methods of using real-time social media technologies for detection and remote monitoring of hiv outcomes," *Preventive medicine*, vol. 63, pp. 112–115, 2014.

[74] M. E. Ireland, Q. Chen, H. A. Schwartz, L. H. Ungar, and D. Albarracin, "Action tweets linked to reduced county-level hiv prevalence in the united states: Online messages and structural determinants," *AIDS and Behavior*, vol. 20, no. 6, pp. 1256–1264, 2016.

[75] M. E. Ireland, H. A. Schwartz, Q. Chen, L. H. Ungar, and D. Albarracín, "Future-oriented tweets predict lower county-level hiv prevalence in the united states." *Health Psychology*, vol. 34, no. S, p. 1252, 2015.

[76] J. W. Pennebaker, M. R. Mehl, and K. G. Niederhoffer, "Psychological aspects of natural language use: Our words, our selves," *Annual review of psychology*, vol. 54, no. 1, pp. 547–577, 2003.

[77] L. Zhou, D. Zhang, C. C. Yang, and Y. Wang, "Harnessing social media for health information management," *Electronic commerce research and applications*, vol. 27, pp. 139–151, 2018.

[78] J. Ginsberg, M. H. Mohebbi, R. S. Patel, L. Brammer, M. S. Smolinski, and L. Brilliant, "Detecting influenza epidemics using search engine query data," *Nature*, vol. 457, no. 7232, pp. 1012–1014, 2009.

[79] J. R. Ortiz, H. Zhou, D. K. Shay, K. M. Neuzil, A. L. Fowlkes, and C. H. Goss, "Monitoring influenza activity in the united states: a comparison of traditional surveillance systems with google flu trends," *PloS one*, vol. 6, no. 4, p. e18687, 2011.

[80] H. Achrekar, A. Gandhe, R. Lazarus, S.-H. Yu, and B. Liu, "Predicting flu trends using twitter data," in *2011 IEEE conference on computer communications workshops (INFOCOM WKSHPS)*. IEEE, 2011, pp. 702–707.

[81] M. Smith, D. A. Broniatowski, M. J. Paul, and M. Dredze, "Towards real-time measurement of public epidemic awareness: Monitoring influenza awareness through twitter," in *AAAI Spring Symposium on Observational Studies through Social Media and Other Human-Generated Content*, 2016.

[82] J. J. Klembczyk, M. Jalalpour, S. Levin, R. E. Washington, J. M. Pines, R. E. Rothman, and A. F. Dugas, "Google flu trends spatial variability validated against emergency department influenza-related visits," *Journal of medical Internet research*, 2016.

[83] H. Hassan and A. Menezes, "Social text normalization using contextual graph random walks," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2013, pp. 1577–1586.

[84] S. Dutta, T. Saha, S. Banerjee, and S. K. Naskar, "Text normalization in code-mixed social media text," in *2015 IEEE 2nd International Conference on Recent Trends in Information Systems (ReTIS)*. IEEE, 2015, pp. 378–382.

[85] I. Lourentzou, K. Manghnani, and C. Zhai, "Adapting sequence to sequence models for text normalization in social media," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 13, 2019, pp. 335–345.

[86] M. Chan, S. Lohmann, A. Morales, C. Zhai, L. Ungar, D. Holtgrave, and D. Albarracín, "An online risk index for the cross-sectional prediction of new hiv chlamydia, and gonorrhea diagnoses across us counties and across years." *AIDS and behavior*, 2018.

[87] H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, L. Dziurzynski, S. M. Ramones, M. Agrawal, A. Shah, M. Kosinski, D. Stillwell, M. E. Seligman et al., "Personality, gender, and age in the language of social media: The open-vocabulary approach," *PloS one*, vol. 8, no. 9, p. e73791, 2013.

[88] S. Dhoju, M. Main Uddin Rony, M. Ashad Kabir, and N. Hassan, "Differences in health news from reliable and unreliable media," in *Companion of The Web Conf*, 2019.

[89] S. Castelo, T. Almeida, A. Elghafari, A. Santos, K. Pham, E. Nakamura, and J. Freire, "A topic-agnostic approach for identifying fake news pages," in *Companion of The Web Conf*, 2019.

[90] A. E. Fard and S. Lingeswaran, "Misinformation battle revisited: Counter strategies from clinics to artificial intelligence," in *Companion of The Web Conf*, 2020.

[91] G. Park, H. A. Schwartz, J. C. Eichstaedt, M. L. Kern, M. Kosinski, D. J. Stillwell, L. H. Ungar, and M. E. Seligman, "Automatic personality assessment through social media language." *Journal of personality and social psychology*, vol. 108, no. 6, p. 934, 2015.

[92] T. S. Tomeny, C. J. Vargo, and S. El-Toukhy, "Geographic and demographic correlates of autism-related anti-vaccine beliefs on twitter, 2009-15," *Social science & medicine*, vol. 191, pp. 168–175, 2017.

[93] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, "Learning under concept drift: A review," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 12, pp. 2346–2363, 2018.

[94] E. Gabrilovich, S. Markovitch et al., "Computing semantic relatedness using wikipedia-based explicit semantic analysis." in *IJcAI*, vol. 7, 2007, pp. 1606–1611.

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

[96] L. Li, B. Qin, W. Ren, and T. Liu, "Document representation and feature combination for deceptive spam review detection," *Neurocomputing*, vol. 254, pp. 33–41, 2017.

[97] S. Massung and C. Zhai, "Syntacticdiff: Operator-based transformation for comparative text mining," in *2015 IEEE International Conference on Big Data (Big Data)*. IEEE, 2015, pp. 571–580.

[98] D. Jackson, D. A. Ladd et al., "Semantic diff: A tool for summarizing the effects of modifications." in *ICSM*, vol. 94. Citeseer, 1994, pp. 243–252.

[99] C. Zhai, A. Velivelli, and B. Yu, "A cross-collection mixture model for comparative text mining," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, 2004, pp. 743–748.

[100] I. Titov and R. McDonald, "Modeling online reviews with multi-grain topic models," in *Proceedings of the 17th international conference on World Wide Web*, 2008, pp. 111–120.

[101] H. Wang, Y. Lu, and C. Zhai, "Latent aspect rating analysis on review text data: a rating regression approach," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2010, pp. 783–792.

[102] G. Ritchie, "Can computers create humor?" *AI Magazine*, vol. 30, no. 3, p. 71, 2009.

[103] R. Mihalcea and C. Strapparava, "Learning to laugh (automatically): Computational models for humor recognition," *Computational Intelligence*, vol. 22, no. 2, pp. 126–142, 2006.

[104] A. Reyes, P. Rosso, and D. Buscaldi, "From humor recognition to irony detection: The figurative language of social media," *Data & Knowledge Engineering*, vol. 74, pp. 1–12, 2012.

[105] A. Reyes, M. Potthast, P. Rosso, and B. Stein, "Evaluating humour features on web comments." in *LREC*, 2010.

[106] S. Attardo, *Linguistic theories of humor*. Walter de Gruyter, 1994, vol. 1.

[107] J. Pearl, *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Elsevier, 2014.

[108] C. Glymour, K. Zhang, and P. Spirtes, "Review of causal discovery methods based on graphical models," *Frontiers in genetics*, vol. 10, p. 524, 2019.

[109] Q. Mei and C. Zhai, "A note on em algorithm for probabilistic latent semantic analysis."

[110] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[111] Y. Lu, Q. Mei, and C. Zhai, "Investigating task performance of probabilistic topic models: an empirical study of plsa and lda," *Information Retrieval*, vol. 14, no. 2, pp. 178–203, 2011.

[112] R. Mihalcea and S. Pulman, "Characterizing humour: An exploration of features in humorous texts," in *Computational Linguistics and Intelligent Text Processing*. Springer, 2007, pp. 337–347.

[113] L. Ratinov, D. Roth, D. Downey, and M. Anderson, "Local and global algorithms for disambiguation to wikipedia," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 1375–1384.

[114] B.-J. Hsu and J. Glass, "Iterative language model estimation: efficient data structure & algorithms," in *Proceedings of Interspeech*, vol. 8, 2008, pp. 1–4.

[115] J. Su, A. Sharma, and S. Goel, "The effect of recommendations on network structure," in *Proceedings of the 25th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2016, pp. 1157–1167.

[116] N. Rizzolo and D. Roth, "Learning based java for rapid development of nlp systems." in *LREC*, 2010.

[117] J. A. Paulos, *Mathematics and humor: A study of the logic of humor*. University of Chicago Press, 2008.

[118] K. Binsted, A. Nijholt, O. Stock, C. Strapparava, G. Ritchie, R. Manurung, H. Pain, A. Waller, and D. O'Mara, "Computational humor," *Intelligent Systems, IEEE*, vol. 21, no. 2, pp. 59–69, 2006.

[119] G. Lessard and M. Levison, "Computational modelling of linguistic humour: Tom swifties," in *ALLC/ACH Joint Annual Conference, Oxford*, 1992, pp. 175–178.

[120] R. Manurung, G. Ritchie, H. Pain, A. Waller, D. O'Mara, and R. Black, "The construction of a pun generator for language skills development," *Applied Artificial Intelligence*, vol. 22, no. 9, pp. 841–869, 2008.

[121] J. McKay, "Generation of idiom-based witticisms to aid second language learning," *Stock et al.(2002)*, pp. 77–87, 2002.

[122] O. Stock and C. Strapparava, "Hahacronym: Humorous agents for humorous acronyms," *Stock, Oliviero, Carlo Strapparava, and Anton Nijholt. Eds*, pp. 125–135, 2002.

[123] P.-Y. Chen and V.-W. Soo, "Humor recognition using deep learning," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018, pp. 113–117.

[124] B. C. Wallace, "Computational irony: A survey and new perspectives," *Artificial Intelligence Review*, pp. 1–17, 2013.

[125] N. Hossain, J. Krumm, M. Gamon, and H. Kautz, "Semeval-2020 task 7: Assessing humor in edited news headlines (to appear)," in *Proceedings of the 14th International Workshop on Semantic Evaluation*, 2020.

[126] C. Kiddon and Y. Brun, "That's what she said: double entendre identification," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*. Association for Computational Linguistics, 2011, pp. 89–94.

[127] A. P. McGraw, C. Warren, and C. Kan, "Humorous complaining," *Journal of Consumer Research*, vol. 41, no. 5, pp. 1153–1171, 2015.

[128] V. Blinov, V. Bolotova-Baranova, and P. Braslavski, "Large dataset and language model fun-tuning for humor recognition," in *Proceedings of the 57th annual meeting of the association for computational linguistics*, 2019, pp. 4027–4032.

[129] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proceedings of the 2008 International Conference on Web Search and Data Mining*. ACM, 2008, pp. 219–230.

[130] H. Sun, A. Morales, and X. Yan, "Synthetic review spamming and defense," in *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2013, pp. 1088–1096.

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

[132] Y. Ma, G. Chen, and Q. Wei, "Finding users preferences from large-scale online reviews for personalized recommendation," *Electronic Commerce Research*, vol. 17, no. 1, pp. 3–29, 2017.

[133] Y. Zhang, M. Chen, D. Huang, D. Wu, and Y. Li, "idoctor: Personalized and professionalized medical recommendations based on hybrid matrix factorization," *Future Generation Computer Systems*, vol. 66, pp. 30–35, 2017.

[134] Q. Li, Y. Li, J. Gao, B. Zhao, W. Fan, and J. Han, "Resolving conflicts in heterogeneous data by truth discovery and source reliability estimation," in *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*. ACM, 2014, pp. 1187–1198.

[135] L. Yang, M. Qiu, S. Gottipati, F. Zhu, J. Jiang, H. Sun, and Z. Chen, "Cqarank: jointly model topics and expertise in community question answering," in *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. ACM, 2013, pp. 99–108.

[136] Y. Li, N. Du, C. Liu, Y. Xie, W. Fan, Q. Li, J. Gao, and H. Sun, "Reliable medical diagnosis from crowdsourcing: Discover trustworthy answers from non-experts," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*. ACM, 2017, pp. 253–261.

[137] J. Roozenbeek and S. van der Linden, "Fake news game confers psychological resistance against online misinformation," *Palgrave Communications*, vol. 5, no. 1, pp. 1–10, 2019.

[138] A. Dharawat, I. Lourentzou, A. Morales, and C. Zhai, "Drink bleach or do what now? covid-hera: A dataset for risk-informed health decision making in the presence of covid19 misinformation," 2020.

[139] Y. Alfasi, "The grass is always greener on my friends' profiles: The effect of facebook social comparison on state self-esteem and depression," *Personality and Individual Differences*, vol. 147, pp. 111–117, 2019.

[140] Y. Wang, J. Luo, R. Niemi, and Y. Li, "To follow or not to follow: Analyzing the growth patterns of the trumpists on twitter," *arXiv preprint arXiv:1603.08174*, 2016.

[141] G. Amati, S. Angelini, F. Capri, G. Gambosi, G. Rossi, and P. Vocca, "On the retweet decay of the evolutionary retweet graph," in *International Conference on Smart Objects and Technologies for Social Good*. Springer, 2016, pp. 243–253.

[142] P. Zola, G. Cola, M. Mazza, and M. Tesconi, "Interaction strength analysis to model retweet cascade graphs," 2020.

[143] R. Vijayan and G. Mohler, "Forecasting retweet count during elections using graph convolution neural networks," in *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 2018, pp. 256–262.

[144] D. R. Bild, Y. Liu, R. P. Dick, Z. M. Mao, and D. S. Wallach, "Aggregate characterization of user behavior in twitter and analysis of the retweet graph," *ACM Transactions on Internet Technology (TOIT)*, vol. 15, no. 1, pp. 1–24, 2015.

[145] C. Wilson, A. Sala, K. P. Puttaswamy, and B. Y. Zhao, "Beyond social graphs: User interactions in online social networks and their implications," *ACM Transactions on the Web (TWEB)*, vol. 6, no. 4, pp. 1–31, 2012.

[146] K. Pelechrinis, V. Zadorozhny, and V. Oleshchuk, "A cognitive-based scheme for user reliability and expertise assessment in q&a social networks," in *2011 IEEE International Conference on Information Reuse & Integration*. IEEE, 2011, pp. 545–550.

[147] P. Nakov, D. Hoogeveen, L. Màrquez, A. Moschitti, H. Mubarak, T. Baldwin, and K. Verspoor, "Semeval-2017 task 3: Community question answering," in *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, 2017, pp. 27–48.

[148] E. Gilbert, "Widespread underprovision on reddit," in *Proceedings of the 2013 conference on Computer supported cooperative work*. ACM, 2013, pp. 803–808.

[149] B. Zhao and J. Han, "A probabilistic model for estimating real-valued truth from conflicting sources," 2012.

[150] X. Yin, J. Han, and P. S. Yu, "Truth discovery with multiple conflicting information providers on the web," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '07. New York, NY, USA: ACM, 2007. [Online]. Available: http://doi.acm.org/10.1145/1281192.1281309 pp. 1048–1052.

[151] J. Pasternack and D. Roth, "Making better informed trust decisions with generalized fact-finding," in *IJCAI*, 2011.

[152] A. Galland, S. Abiteboul, A. Marian, and P. Senellart, "Corroborating information from disagreeing views," in *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, ser. WSDM '10. New York, NY, USA: ACM, 2010. [Online]. Available: http://doi.acm.org/10.1145/1718487.1718504 pp. 131–140.

[153] X. L. Dong, L. Berti-Equille, and D. Srivastava, "Integrating conflicting data: The role of source dependence," *Proc. VLDB Endow.*, vol. 2, no. 1, pp. 550–561, Aug. 2009. [Online]. Available: https://doi.org/10.14778/1687627.1687690

[154] M. Wan, X. Chen, L. Kaplan, J. Han, J. Gao, and B. Zhao, "From truth discovery to trustworthy opinion discovery: An uncertainty-aware quantitative modeling approach," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2016, pp. 1885–1894.

[155] H. Zhang, Y. Li, F. Ma, J. Gao, and L. Su, "Texttruth: An unsupervised approach to discover trustworthy information from multi-sourced text data," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 2729–2737.

[156] E. Agichtein, C. Castillo, D. Donato, A. Gionis, and G. Mishne, "Finding high-quality content in social media," in *WSDM*, 2008.

[157] A. Barrón-Cedeno, S. Filice, G. Da San Martino, S. R. Joty, L. Màrquez, P. Nakov, and A. Moschitti, "Thread-level information for comment classification in community question answering." in *ACL (2)*, 2015, pp. 687–693.

[158] T. Mihaylova, G. Karadzhov, P. Atanasova, R. Baly, M. Mohtarami, and P. Nakov, "Semeval-2019 task 8: Fact checking in community question answering," in *Proceedings of the 13th International Workshop on Semantic Evaluation (SemEval-2019)*, 2019, pp. 856–865.

[159] Z. Wang, H. Mi, and A. Ittycheriah, "Sentence similarity learning by lexical decomposition and composition," *arXiv preprint arXiv:1602.07019*, 2016.

[160] D. P. Bertsekas, *Nonlinear programming*. Athena scientific Belmont, 1999.

[161] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[162] S. Lyu, W. Ouyang, Y. Wang, H. Shen, and X. Cheng, "What we vote for? answer selection from user expertise view in community question answering," in *The World Wide Web Conference*, ser. WWW '19. New York, NY, USA: ACM, 2019. [Online]. Available: http://doi.acm.org/10.1145/3308558.3313510 pp. 1198–1209.

[163] J. Wen, J. Ma, Y. Feng, and M. Zhong, "Hybrid attentive answer selection in cqa with deep users modelling," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[164] Q. Li, Y. Li, J. Gao, L. Su, B. Zhao, M. Demirbas, W. Fan, and J. Han, "A confidence-aware approach for truth discovery on long-tail data," *Proceedings of the VLDB Endowment*, vol. 8, no. 4, pp. 425–436, 2014.

[165] T. Mihaylova, P. Nakov, L. Marquez, A. Barron-Cedeno, M. Mohtarami, G. Karadzhov, and J. Glass, "Fact checking in community forums," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[166] H.-J. Oh, Y.-C. Yoon, and H. K. Kim, "Finding more trustworthy answers: Various trustworthiness factors in question answering," *Journal of Information Science*, vol. 39, no. 4, pp. 509–522, 2013.

[167] C. Lioma, B. Larsen, W. Lu, and Y. Huang, "A study of factuality, objectivity and relevance: three desiderata in large-scale information retrieval?" in *Proceedings of the 3rd IEEE/ACM International Conference on Big Data Computing, Applications and Technologies*. ACM, 2016, pp. 107–117.

[168] Y. Li, Q. Li, J. Gao, L. Su, B. Zhao, W. Fan, and J. Han, "On the discovery of evolving truth," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 675–684.

[169] Y. Zheng, G. Li, Y. Li, C. Shan, and R. Cheng, "Truth inference in crowdsourcing: is the problem solved?" *Proceedings of the VLDB Endowment*, vol. 10, no. 5, pp. 541–552, 2017.

[170] Q. Li, F. Ma, J. Gao, L. Su, and C. J. Quinn, "Crowdsourcing high quality labels with a tight budget," in *Proceedings of the ninth acm international conference on web search and data mining*. ACM, 2016, pp. 237–246.

[171] T. Mukherjee, B. Parajuli, P. Kumar, and E. Pasiliao, "Truthcore: Non-parametric estimation of truth from a collection of authoritative sources," in *Big Data (Big Data), 2016 IEEE International Conference on*. IEEE, 2016, pp. 976–983.

[172] V. Vydiswaran, C. Zhai, and D. Roth, "Content-driven trust propagation framework," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 974–982.

[173] Y. Li, J. Gao, C. Meng, Q. Li, L. Su, B. Zhao, W. Fan, and J. Han, "A survey on truth discovery," *ACM Sigkdd Explorations Newsletter*, vol. 17, no. 2, pp. 1–16, 2016.

[174] F. Ma, Y. Li, Q. Li, M. Qiu, J. Gao, S. Zhi, L. Su, B. Zhao, H. Ji, and J. Han, "Faitcrowd: Fine grained truth discovery for crowdsourced data aggregation," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2015, pp. 745–754.

[175] R. Y. K. Lau, Y. Xia, and Y. Ye, "A probabilistic generative model for mining cybercriminal networks from online social media," *IEEE Comp. Int. Mag.*, vol. 9, no. 1, pp. 31–43, 2014.

[176] T. Rao and S. Srivastava, "Analyzing stock market movements using twitter sentiment analysis," in *Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012)*, ser. ASONAM '12, 2012, pp. 119–123.

[177] C. Zhai and S. Massung, *Text Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining*. Morgan & Claypool and ACM, 2016.

[178] D. Alvarez-Melis and M. Saveski, "Topic modeling in twitter: Aggregating tweets by conversations." in *ICWSM*, 2016, pp. 519–522.

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

[180] "Sexually transmitted disease surveillance 2016," *Centers for Desease Controll and Prevention, Atlanta, US Department*, 2017.

[181] S. D. Young, "Behavioral insights on big data: using social media for predicting biomedical outcomes," *Trends in microbiology*, vol. 22, no. 11, pp. 601–602, 2014.

[182] Z. S. Harris, "Distributional structure," *¡i¿WORD¡/i¿*, vol. 10, no. 2-3, pp. 146–162, 1954. [Online]. Available: https://doi.org/10.1080/00437956.1954.11659520

[183] B. Han, A. Hugo, A. Rahimi, L. Derczynski, and T. Baldwin, "Twitter geolocation prediction shared task of the 2016 workshop on noisy user-generated text," *WNUT 2016*, p. 213, 2016.

[184] B. Han, P. Cook, and T. Baldwin, "Text-based twitter user geolocation prediction," *Journal of Artificial Intelligence Research*, pp. 451–500, 2014.

[185] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine learning*, vol. 63, no. 1, pp. 3–42, 2006.

[186] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the eleventh annual conference on Computational learning theory*, 1998, pp. 92–100.

[187] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *33rd annual meeting of the association for computational linguistics*, 1995, pp. 189–196.

[188] S. Bickel and T. Scheffer, "Multi-view clustering." in *ICDM*, vol. 4, no. 2004, 2004, pp. 19–26.

[189] T. Xia, D. Tao, T. Mei, and Y. Zhang, "Multiview spectral embedding," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 6, pp. 1438–1446, 2010.

[190] J. Yu, D. Tao, Y. Rui, and J. Cheng, "Pairwise constraints based multiview features fusion for scene classification," *Pattern Recognition*, vol. 46, no. 2, pp. 483–496, 2013.

[191] B. Xie, Y. Mu, D. Tao, and K. Huang, "m-sne: Multiview stochastic neighbor embedding," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 41, no. 4, pp. 1088–1096, 2011.

[192] J. C. Eichstaedt, H. A. Schwartz, M. L. Kern, G. Park, D. R. Labarthe, R. M. Merchant, S. Jha, M. Agrawal, L. A. Dziurzynski, M. Sap et al., "Psychological language on twitter predicts county-level heart disease mortality," *Psychological science*, vol. 26, no. 2, pp. 159–169, 2015.

[193] J. Jashinsky, S. H. Burton, C. L. Hanson, J. West, C. Giraud-Carrier, M. D. Barnes, and T. Argyle, "Tracking suicide risk factors through twitter in the us," *Crisis*, 2015.

[194] A. Signorini, A. M. Segre, and P. M. Polgreen, "The use of twitter to track levels of disease activity and public concern in the us during the influenza a h1n1 pandemic," *PloS one*, vol. 6, no. 5, p. e19467, 2011.

[195] A. A. Aslam, M.-H. Tsou, B. H. Spitzberg, L. An, J. M. Gawron, D. K. Gupta, K. M. Peddecord, A. C. Nagel, C. Allen, J.-A. Yang et al., "The reliability of tweets as a supplementary method of seasonal influenza surveillance," *Journal of medical Internet research*, vol. 16, no. 11, p. e250, 2014.

[196] J. C. Santos and S. Matos, "Analysing twitter and web queries for flu trend prediction," *Theoretical Biology and Medical Modelling*, vol. 11, no. Suppl 1, p. S6, 2014.

[197] S. Ram, W. Zhang, M. Williams, and Y. Pengetnze, "Predicting asthma-related emergency department visits using big data," 2015.

[198] H. Gu, B. Chen, H. Zhu, T. Jiang, X. Wang, L. Chen, Z. Jiang, D. Zheng, and J. Jiang, "Importance of internet surveillance in public health emergency control and prevention: evidence from a digital epidemiologic study during avian influenza a h7n9 outbreaks," *Journal of medical Internet research*, vol. 16, no. 1, 2014.

[199] C. L. Hanson, S. H. Burton, C. Giraud-Carrier, J. H. West, M. D. Barnes, and B. Hansen, "Tweaking and tweeting: exploring twitter for nonmedical use of a psychostimulant drug (adderall) among college students," *Journal of medical Internet research*, vol. 15, no. 4, 2013.

[200] J. Olsen, "Infodemiology to improve public health situational awareness: An investigation of 2010 pertussis outbreaks in california, michigan and ohio," Ph.D. dissertation, THE UNIVERSITY OF NORTH CAROLINA AT CHAPEL HILL, 2013.

[201] H. Sueki, "The association of suicide-related twitter use with suicidal behaviour: A cross-sectional study of young internet users in japan," *Journal of affective disorders*, vol. 170, pp. 155–160, 2015.

[202] R. Nagar, Q. Yuan, C. C. Freifeld, M. Santillana, A. Nojima, R. Chunara, and J. S. Brownstein, "A case study of the new york city 2012-2013 influenza season with daily geocoded twitter data from temporal and spatiotemporal perspectives," *Journal of medical Internet research*, vol. 16, no. 10, 2014.

[203] A. Ritter, C. Cherry, and B. Dolan, "Unsupervised modeling of twitter conversations," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 172–180.

[204] M. J. Paul and M. Dredze, "A model for mining public health topics from twitter," *Health*, vol. 11, pp. 16–6, 2012.

[205] J. Chang, S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei, "Reading tea leaves: How humans interpret topic models," in *Advances in neural information processing systems*, 2009, pp. 288–296.

[206] J. Chen and D. Yang, "Multi-view sequence-to-sequence models with conversational structure for abstractive dialogue summarization," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-main.336 pp. 4106–4118.