Mining Opinions and Trends from Social Media

Amit Gupta
LSIR, I&C, EPFL

Abstract—The popularity of web and social media has resulted in tremendous growth in the amount of natural language data being generated everyday. Since the size of data is huge, it is not feasible to perform manual analysis and hence there has been a lot of interest in developing novel techniques to automatically analyze text data and discover hidden knowledge. In this report, we discuss sentiment analysis or opinion mining which is a sub-problem of text analysis and aims to detect subjective information such as opinions, feelings and attitudes from text. We describe three different approaches for sentiment analysis that belong to different paradigms of machine learning i.e. unsupervised, supervised and ontology-based learning. We also provide a comparison between the approaches and discuss their benefits and shortcomings. Towards the end, we briefly describe our research direction and plans for the near future.


I. INTRODUCTION

SOCIAL media sites are internet sites such as blogs, social networks, microblogs, wikis etc. where users interact freely and share and express information about their own lives, other people’s lives and worldly matters. Some examples of social media sites include Facebook, Twitter and Wikipedia.

Proposal submitted to committee: July 17, 2014; Candidacy exam date: July 24, 2014; Candidacy exam committee: Prof. Matthias Grossglauser, Prof. Karl Aberer, Prof. Jean-Cédric Chappelier.

This research plan has been approved:

Date: ________________________________

Doctoral candidate: ________________________________
(name and signature)

Thesis director: ________________________________
(name and signature)

Thesis co-director: ________________________________
(if applicable) (name and signature)

Doct. prog. director: ________________________________
(B. Falsafi) (signature)

With the onset of Web 2.0, the popularity and reach of these sites has exploded. For example, the number of active users on Facebook and Twitter, two of the most popular social media sites, has increased by more than 700% in the last few years [1][2].

Social media sites allow users to constantly generate huge amounts of data using a mix of natural language, pictures, photos and audio. This data is extremely valuable and contains important actionable information and insights which are useful from an economic as well as a research point of view. Since the size of data is huge and most of it is generated entirely for human consumption, it poses a significant challenge to create a programmatic access to this information. In the last few years, there has been a lot of research in problems related to information extraction from the natural language component of social media data. Main examples of such problems include entity recognition, sentiment analysis, opinion mining, subjectivity analysis and spam detection.

In this report, we focus on the problem of sentiment analysis which aims to determine the attitude or opinion of the writer towards certain topic or the contextual polarity of the text. We describe three different approaches that are frequently applied to its resolution. First, we describe joint sentiment-topic model (JST) [3] which employs an unsupervised approach for document-level sentiment classification. Second, we discuss a supervised approach using minimum cuts-based classification algorithm as described in [4]. Finally, we describe an ontology-based approach for sentiment analysis of Twitter posts as proposed in [5]. Ontology-based approaches can be loosely classified as distant supervision techniques.

These approaches belong to different ideologies of machine learning and hence cover a good breadth of the nature of research done in this area. We also compare the three approaches and discuss the advantages and disadvantages for each of them. Later, we describe our research proposal and discuss how we aim to combine the above approaches and create a hybrid solution.

The rest of the report is organized as follows. Sections II,III and IV describe the three approaches. Section V provides a comparison between the approaches and discusses how to choose a suitable one for a given situation. Section VI describes our research proposal. Section VII concludes the report and outlines the future work.

II. UNSUPERVISED LEARNING FOR SENTIMENT ANALYSIS

In this section, we discuss an unsupervised model for sentiment analysis also known as joint sentiment-topic model (JST). As the name suggests, JST performs simultaneous detection of topics and sentiments from text documents. JST
is an extension of Latent Dirichlet Allocation (LDA) \cite{Blei2003}, a highly cited and frequently used topic model. We first describe LDA and JST briefly and then discuss the results of JST for sentiment analysis.

A. Latent Dirichlet Allocation (LDA)

Latent Dirichlet Allocation is a generative probabilistic model which can be used to detect topics from a collection of text documents. It is a three-level hierarchical bayesian model, which makes the assumption that each document is a finite mixture of topics and each topic is a probability distribution over words. The generative process of LDA is described as follows.

- Choose document-specific topic distribution \( \theta \sim \text{Dir}(\alpha) \)
- Choose word distributions for each topic \( \phi_k \sim \text{Dir}(\beta) \)
- For each word \( w_i \) in each document
  - Assign a topic \( z_i \sim \text{Multinomial}(\theta) \)
  - Draw a word \( w_i \sim \phi_{z_i} \)

Here, \( \alpha \) and \( \beta \) are initial parameters for Dirichlet function (Dir). The plate diagram is shown in Figure 1. As is clear from the generative process, LDA completely ignores the order of words semantics for document modeling. Learning the various probability distributions. In other words, LDA follows bag-of-words semantics for document modeling. Learning the various probability distributions from a given set of text documents is a problem of Bayesian inference and has a variety of different solutions including Variational Bayes \cite{Blei2003}, Gibbs Sampling \cite{Gibbs1993} and Expectation Propagation \cite{Minka2001}. We briefly describe the Gibbs Sampling approach, as JST also employs this approach for inference.

\[ P(\mathbf{z_i} = j | \mathbf{z_{-i}}, \mathbf{w}) \text{ is the probability of topic for word } w_i \text{ being } j \text{ given current topic assignments for other words} \]
\[ n_{w_i,j} \text{ is count of } w_i \text{ assigned to topic } j \]
\[ n_{-i,j} \text{ is count of any word assigned to topic } j \]
\[ n_{d_{-i,j}} \text{ is count of topic } j \text{ in document } d_i \]
\[ n_{d_{i,j}} \text{ is the number of words in document } d_i \]

Here, \( -i \) subscript means that the current assignment is excluded from the calculation. This formula has a very intuitive explanation as the first ratio represents the probability of drawing \( w_i \) from topic \( j \) while the second ratio represents the probability of drawing topic \( j \) from document \( d_i \).

C. JST Model Description

JST extends LDA by adding another layer for sentiment modeling. The plate for JST is shown in Figure 2. The generative process for JST is as follows.

- Choose document-specific sentiment distribution \( \pi_d \sim \text{Dir}(\gamma) \)
- Choose document,sentiment-specific topic distributions \( \theta_{d,l} \sim \text{Dir}(\alpha) \)
- Choose sentiment,topic-specific word distributions \( \phi_{z,l} \sim \text{Dir}(\beta) \)
- For each word \( w_i \) in each document
  - Assign a topic \( l_i \sim \text{Multinomial}(\pi_d) \)
  - Assign a topic \( z_i \sim \text{Multinomial}(\theta_{d,l_i}) \)
  - Choose a word \( w_i \sim \phi_{z_i,l_i} \)

\[ P(\mathbf{z_i} = j | \mathbf{l_i}, \mathbf{z_{-i}}, \mathbf{w}, \alpha, \beta, \gamma) = \frac{\sum_k \theta_{d_k,j} \pi_{d_k} \alpha \phi_{z_i,k} \beta}{\sum_k \theta_{d_k,j} \pi_{d_k} \alpha \phi_{z_i,k} \beta + \gamma} \]
where \( \mathbf{w} \) is the entire corpus
\( \mathbf{z_i} \) are topic assignments
\( P(\mathbf{z_i} = j | \mathbf{z_{-i}}, \mathbf{w}) \) is the probability of topic for word \( w_i \)
\( \text{being } j \text{ given current topic assignments for other words} \)
\( n_{w_i,j} \text{ is count of } w_i \text{ assigned to topic } j \)
\( n_{-i,j} \text{ is count of any word assigned to topic } j \)
\( n_{d_{-i,j}} \text{ is count of topic } j \text{ in document } d_i \)
\( n_{d_{i,j}} \text{ is the number of words in document } d_i \)

D. Gibbs Sampling for JST

Similar to LDA, Gibbs Sampling training equation for JST is as follows.

\[ P(\mathbf{z_i} = j, \mathbf{l_i} = k | \mathbf{z_{-i}}, \mathbf{l_{-i}}, \mathbf{w}, \alpha, \beta, \gamma) \propto P(\mathbf{l_i} = k | \mathbf{z_{-i}}, \mathbf{l_{-i}}, \mathbf{w}, \alpha, \beta, \gamma) \]
\[ \times P(\mathbf{z_i} = j | \mathbf{l_i} = k, \mathbf{z_{-i}}, \mathbf{w}, \alpha, \beta, \gamma) \]
\[ \times P(\mathbf{w_i} | \mathbf{z_i} = j, \mathbf{l_i} = k, \mathbf{z_{-i}}, \mathbf{w}, \alpha, \beta, \gamma) \]
\[ \text{where } P(\mathbf{z_i} = j | \mathbf{l_i} = k | \mathbf{z_{-i}}, \mathbf{w}, \alpha, \beta, \gamma) \text{ is the probability of word } w_i \text{ being assigned topic } j, \text{ sentiment label } k \text{ given other topic,sentiment assignments} \]
$n_{d,k}^{-t}$ is the count of sentiment $k$ assigned in document $d$.

$n_{d}^{-t}$ is the count of words in document $d$.

$n_{d,k,j}^{-t}$ is the count of words assigned to sentiment $k$ and topic $j$ in document $d$.

$n_{d,k}^{-t}$ is the count of words assigned to sentiment $k$ in document $d$.

$n_{k,j}^{-t}$ is count of $w_i$ assigned to sentiment $k$ and topic $j$.

$n_{k,j}^{-t}$ is count of any word assigned to sentiment $k$ and topic $j$.

Similar to LDA, the training equation for JST can also be explained intuitively. The first ratio represents the probability of sentiment $k$ in document $d$, second ratio represents probability of topic $j$ in document $d$ and sentiment $k$, and the third ratio represents the probability of drawing $w_i$ from sentiment $k$ and topic $j$.

### E. Intuitive Explanation of LDA and JST

Since, LDA and JST follow bag-of-words semantics, each document in the collection can be represented as a vector in word space. Thus, a collection of documents forms a matrix, i.e., the document-word matrix. Intuitively speaking, LDA decomposes the document-word matrix into a product of two smaller matrices, a document-topic matrix and a topic-word matrix. The initial parameters $\alpha$ and $\beta$ determine the sparsity of the two matrices. In the most general setting, LDA tries to keep the matrices as sparse as possible by keeping fewer topics per documents and fewer words per topic. Similarly, JST decomposes the document-word matrix into a product of three sparse matrices, a document-sentiment matrix, a sentiment-topic matrix and 3-dimensional sentiment-topic word matrix. Figure 3 explains this intuition in a graphical fashion.

### F. Experimental Results for JST

In [3], authors report the performance of JST for the task of document-level sentiment classification on movie reviews dataset [9]. To improve the accuracy of sentiment detection, the authors incorporate a subjectivity lexicon into JST priors. For this, it sets the initial values of sentiment-topic word distribution using the sentiment labels found in the lexicon. Overall, the model performs very close to the state-of-the-art supervised approaches and achieves approximately 85% accuracy with the use of subjectivity lexicon. This is a significant achievement since JST is completely unsupervised and does not require labeled data for training purposes.

### III. SUPERVISED LEARNING FOR SENTIMENT ANALYSIS

In contrast to the unsupervised approach mentioned in previous section, supervised approaches assume the presence of training data labeled with sentiment polarities such as positive, negative, neutral etc. Most of the research in supervised learning for sentiment analysis involves a set of features such as unigrams, POS tags, position of words etc. and a machine learning classifier such as SVM or Naive Bayes (NB). In this section, we describe a supervised approach for document-level sentiment classification which is discussed in [4].

#### A. Overall Approach

The authors hypothesize that objective sentences in a document reduce the accuracy of polarity classifiers and hence, propose a two-step solution for document-level sentiment classification:

1) **Subjectivity Detection Step**: Use a standard sentence-level subjectivity classifier (such as NB or SVM) along with minimum cuts algorithm as described in Section III-B to select most subjective sentences from a given document.

2) **Polarity Classification Step**: Use a standard document-level polarity classifier on features generated only from the subjective sentences and not the whole document.

In both steps, following [10], this paper uses unigram-presence features for classification: the $i$th coordinate of feature is 1 if the corresponding unigram occurs in input, 0 otherwise. For training the subjectivity classifier, a dataset of 10,000 subjective and objective sentences is used. For training the polarity classifier, a dataset of 2,000 positive and negative movie reviews is used.

#### B. Subjectivity Detection

Authors model the problem of sentence-level subjectivity classification as the problem of finding a minimum cut in a specifically designed undirected graph. The graph is described as follows:

- Add $n$ vertices $v_1, v_2, \ldots, v_n$ one for each sentence.
- Add a source vertex $s$ and a sink vertex $t$.
- For each sentence $x_i$, add $n$ edges $(s, v_i)$ with Individual subjectivity scores $\text{ind}_{\text{sub}}(x_i)$ computed using subjectivity classifier.
- For each sentence $x_i$, add $n$ edges $(v_i, t)$ with Individual objectivity scores $\text{ind}_{\text{obj}}(x_i)$ (same as $(1 - \text{ind}_{\text{sub}}(x_i))$).
- For each pair of sentences $(x_i, x_j)$, add $\binom{n}{2}$ edges $(v_i, v_j)$ with Association scores $\text{assoc}(x_i, x_j)$ between sentences based on proximity to leverage coherence: texts near each other share similar subjectivity status.

One example of such graph is shown in Figure 4. Basic algebra dictates that a minimum cut partition for the above graph will minimize the following quantity:

$$
\sum_{x \in \text{obj}} \text{ind}_{\text{obj}}(x) + \sum_{x \in \text{sub}} \text{ind}_{\text{sub}}(x) + \sum_{x_i \in \text{obj}, x_j \in \text{sub}} \text{assoc}(x_i, x_j)
$$
Intuitively speaking, the above quantity penalizes assigning a wrong class to a sentence (i.e. high \(\text{ind}_{\text{obj}}(x_{\text{sub}})\) or \(\text{ind}_{\text{sub}}(x_{\text{obj}})\)) as well as assigning different classes to tightly-associated sentences \(\text{assoc}(x_{\text{sub}}, x_{\text{obj}})\). This paradigm is quite flexible as the classifier for computing individual subjectivity scores could be modified easily without affecting the cuts-based classification step.

Multiple definitions of proximity functions can be used. The proximity function usually has three parameters: 1) threshold \(T\) which specifies the maximum distance between two sentences which are still considered proximal, 2) a decaying function \(f(d)\) which specifies how the proximity value decays with distance and 3) a constant \(c\) which specifies the relative importance of \(\text{assoc}\) and \(\text{ind}\). Examples of \(f\) include \(e^{1-d}\) and \(\frac{1}{d}\). The assoc score can be defined as:

\[
\text{assoc}(s_i, s_j) = \begin{cases} 
    f(j - i) \times c & \text{if } j - i \leq T \\
    0 & \text{otherwise}
\end{cases}
\]

The paper also experiments with proximity functions which respect paragraph boundaries, i.e. sentences belong to different paragraphs are not proximal.

As discussed in [11], maximum-flow algorithms can be used with near-linear running times to compute the minimum cut in an efficient manner. This is quite appealing as this algorithm finds the exact optimum minimum-cost cut for the graph. This is in contrast to other approaches for classification which use algorithms such as expectation-maximization, gradient descent etc. which frequently get stuck in local minima.

C. Polarity Classification

For document-level polarity classification, \(N\) most subjective sentences are selected from the original document to form a subjective summary. The polarity of original document is computed as the polarity of this summary using a simple polarity classifier as described in section III-A.

D. Experimental Results

In [4], authors report the results for various experiments performed using this approach. Most notably, using NB classifier for subjectivity detection as well as for polarity classification achieves an accuracy of 86.4% which is significantly higher than 82.8% achieved without subjectivity detection. Experiments also show that using polarity classifier on sentences labeled as objective results in an accuracy of only 71%. This clearly shows that the subjectivity detection step discards objective sentences which are much less indicative of overall polarity of document. Experiments also show that smaller values of \(N\) such as 5, also produce good accuracy. This indicates that for the task of document-level polarity classification, just the top five most subjective sentences are almost as informative as the full document.

Lastly, the paper also judges the effect of adding proximity information by comparing \(\text{NB + Prox}\) and \(\text{SVM + Prox}\) with simple \(\text{NB}\) and \(\text{SVM}\) respectively. In both cases, for certain proximity functions, the former give statistically significant improvements over non-proximity counterparts. This indicates that using the proximity of sentences in subjectivity classification does help in improving the overall accuracy of polarity classification.

IV. ONTOLOGY-BASED APPROACHES FOR SENTIMENT ANALYSIS

In this section, we discuss another approach to sentiment analysis as described in [5] which utilizes domain-specific background knowledge-bases i.e. ontologies. Use of ontology enables this approach to perform fine-grained aspect-based sentiment analysis, rather than on a document-level as described in previous approaches.

Ontology is a hierarchical representation of domain knowledge and contains concepts and shared vocabulary to denote the types, properties and interrelationships of concepts. Ontologies provide many benefits in text analysis such as:

- Separate domain knowledge from other knowledge;
- Make domain knowledge reusable and domain-specific assumptions explicit;
- Facilitate a common understanding between domain experts and software agents.

There are three main steps for performing sentiment analysis using an ontology: 1) create an ontology, 2) retrieve documents corresponding to certain concepts and attributes in ontology and 3) compute sentiments from retrieved documents.
A. Ontology Creation

In [5], authors discuss two approaches to ontology creation: 1) learning-based approach and 2) manual approach using formal concept analysis. The accuracy of learning-based approaches for automated ontology extraction is not good and it's still an active area of research. Most of the learning-based approaches such as Ontogen[12] provide an interactive interface for semi-automatic, data-driven ontology development. We now describe Formal Concept Analysis briefly.

Formal Concept Analysis (FCA)

FCA [13] is an iterative manual method for creating an ontology from collection of objects and properties. The basic building blocks of FCA include a set of objects \( O \), a set of attributes \( A \), and a binary incidence relation between \( O \) and \( A \) which define relations of type "object \( o \) has attribute \( a \)". The algorithm to create an ontology using FCA includes following steps:

1) Use seed domain concepts (determined manually) to retrieve an initial set of documents;
2) Manually detect objects and attributes from documents;
3) Populate the incidence table \( I \) using retrieved objects and attributes.

The ontology is further enriched with synonyms and hyponyms of detected attributes using WordNet lexical database [14]. Figure 5 displays an example of ontology for the domain of smartphones built using FCA.

B. Sentiment Analysis

To perform sentiment analysis, first a set of documents are retrieved for each object-attribute pair \((o, a)\) in the \( I \) incidence relation. Then, these documents are sent to a third-party service OpenDover [15] which accepts text documents and ontology as inputs and returns sentiments directed towards the concepts in the ontology. Since OpenDover is not open-source, their sentiment analysis methods are not publicly available and hence not described here.

C. Experimental Results

Figure 6 shows the results of sentiment analysis on tweets related to smartphones using above approach. The actual names of smartphone models are replaced by generic names to maintain neutrality. As you can notice, the sentiments scores are very fine-grained, i.e. they are computed separately for different aspects of smartphones such as battery, processor, display and camera.

This paper also compares the recall ratios of three different retrieval methods: 1) full-fledged semantically enabled ontology based retrieval (SEM); 2) retrieval using ontology without semantic enhancements i.e. synonyms and hyponyms (ONT) and 3) retrieval only using initial seed concepts (CUS). It concludes that recall ratio for SEM is higher than ONT which is significantly higher than CUS.

V. Analysis

In previous sections, we discussed many different approaches for the problem of sentiment analysis. All of these approaches have their own advantages and disadvantages. In this section, we provide a comparison between the three approaches. We also discuss some basic guidelines on how to decide which approach should be applied in a given situation.

A. Comparison of Approaches

Using JST for sentiment analysis has several advantages. First and foremost, it does not require manual labeling of training data, which is usually a very time-consuming and arduous process. Especially in the context of social media analysis, since the size of data is huge, it is very difficult to obtain labeled training data. Secondly, it is completely domain-independent and hence, better equipped for automatically discovery of sentiment and topic words from unseen data. Third, since JST models sentiments and topics simultaneously,
it has the ability to perform fine-grained sentiment analysis for each topic/aspect.

JST also has many disadvantages compared to other approaches for sentiment analysis. First, as discussed before, JST completely ignores the order of words. JST also ignores other semantic information such as similarity between words, synonyms and hyponyms, part-of-speech tags (POS tags), domain knowledge etc. Second, JST and most unsupervised approaches usually have lower accuracy than state-of-the-art supervised approaches. In information retrieval terms, JST and unsupervised approaches in general, produce good recall but lower precision.

In contrast to unsupervised approaches, the main advantage of supervised approaches for sentiment analysis is higher accuracy or precision for a particular domain. Also, since labeled training data is available, information such as POS tags, order of words, and semantic similarity, can be easily incorporated as features for machine learning classifiers. In fact, there is a lot of literature experimenting with different combinations of features and classification algorithms.

The main disadvantage of supervised approaches is the requirement of labeled training data. Supervised approaches are also prone to domain-dependence and over-fitting. Classifiers trained on a training data may not perform well on test data or data from other domains.

Ontology-based approaches attempt to find a middle ground between supervised and unsupervised approaches. In these approaches, unlike supervised approaches, manual effort is required only during ontology creation which is usually less than labeling entire training set. Since ontologies have synonyms, hyponyms and other semantic relationships between the concepts, ontology-based approaches incorporate semantic information during sentiment analysis.

The main disadvantages of ontology-based approaches include the effort in construction of ontology and domain dependence. As discussed before, the task of automated ontology construction is still elusive. Manual creation of ontology requires tremendous efforts from domain experts. There are frequent disagreements between domain experts further increasing the difficulty.

B. How to Choose An Approach?

The question of which approach should be applied for a certain problem depends on many factors such as the type of data, performance requirements, available resources etc.

In theory, if the size of data is small and enough resources are available, supervised approaches should be used as they result in higher accuracy. If it is hard to obtain a manually labeled training set, one of ontology-based approaches or unsupervised approaches should be considered depending on performance requirements. If precision is more important than recall, or if the domain is static with a fixed set of concepts, ontology-based approaches should be used. If the domain is continuously changing such as a twitter stream, unsupervised approaches might perform better and achieve higher recall.

In the context of social media, usually none of the approaches in isolation are sufficient. A hybrid solution combining multiple approaches is usually required to achieve desired precision and recall. In the next section, we discuss our research proposal which describes an example of such hybrid solution.

VI. Research Proposal

In our research, we focus on the domain of climate change which is a popular topic of debate on social media sites such as Twitter, Facebook and blogs. It is also an active area of scientific research and political debates and thousands of related papers and reports are published every year. Hence, it poses a significant challenge to analyze and extract useful information from multiple sources and aggregate them in a meaningful manner. In this section, we describe a variety of research directions we are planning to take in order to effectively tackle this problem.

A. Extension to Debate Analysis

Since the domain of climate change has lots of complex debates, traditional sentiment analysis is not enough to detect the opinions represented in texts. In debates, opinions or arguments are represented as higher-order relations between concepts such as concept-verb-concept tuples. For example, "humans cause climate change." implies that the writer believes humans are causing climate change. However, the sentence contains no sentiment word, hence rendering most of existing sentiment analysis techniques ineffective. To solve this issue, we plan to incorporate semantic information such as POS tags, noun phrases etc. We have already experimented with many algorithms which produce reasonable results on Twitter data.

One such algorithm is described here briefly:

- **Input** - set of tweets
- **Output** - set of concept-verb-concept relations expressing opinions over certain pre-defined issues (known concept-concept pairs)
- **Domain knowledge** - concepts ontology, verbs ontology, negation ontology, issues ontology (known issues)
- **Algorithm steps:**
  1. Clean tweet text to remove random characters, mentions, urls and other noise.
  2. Split cleaned text into sentences.
  3. Generate parse tree for each sentence using Stanford Parser[16].
  4. Detect concepts which only occur as part of noun-phrases in the parse tree.
  5. Detect concept-pairs which belong to issues ontology.
  6. Detect verbs and negation in the text connecting the concept pairs.
  7. Output tuples (concept-verb-concept) or (concept-not verb-concept)

The results for above algorithms are fairly precise but lack reasonable recall. We are still working on it and plan to improve this in future.
B. Hybrid Solution using Crowd-Sourcing

For our project, it is very important to achieve high precision in results. This suggests using a supervised or ontology-based approach for opinion analysis. However, since the domain of climate change is very active and the size of available data is huge, it is not possible to achieve reasonable recall using only supervised or ontology-based approaches. Hence, we adopt a hybrid approach that combines an ontology-based approach along with unsupervised approaches to attain both high precision and recall.

Our proposed approach has following steps:
- Seed ontology creation through manual efforts.
- Precise opinion detection using ontology from continuous data streams.
- Unsupervised approaches to constantly produce candidate concepts and relations for ontology.
- Crowd-sourcing and manual intervention to curate the candidates and add new concepts to ontology.

C. Incorporating Meta-Information

Along with natural language text, social media data contains a variety of meta-information such as personal information of users, location, timestamps, languages, relationships between users, social networks etc. This information can be used to perform many different kinds of analysis such as opinion communities detection, user popularity and influence analysis, credibility analysis and event detection.

Currently, we are working on extending JST to incorporate user-types. The key idea is that there is a finite set of user-types (e.g. climate deniers, politicians etc.) and each user has a specific user-type distribution. During the generation of each word, the model first assigns a user-type and then invokes a user-type,sentiment-specific topic distribution. The rest of the process is same as JST. We have computed Gibbs Sampling training equations for this model, and we plan to implement and test them in the near future.

D. Incorporating Semantic Information in LDA

We plan to extend LDA and enable it to use semantic information from a variety of sources. More specifically, we wish to incorporate two kinds of information: 1) semantic similarity between terms as defined by dictionaries and ontologies and 2) word-level tags as defined by ontologies, POS tagging, existing knowledge bases such as Freebase [17].

The key idea for using semantic similarity between terms is to first transform the vocabulary such that it contains concepts, where a concept is defined as a vector of equivalent terms (for e.g. {climate change, changing climate, global warming etc.}). Afterwards, the generative process of LDA should be modified to first generate a concept for each term, and then generate the term from the concept vector.

The key idea for incorporating word-level tags is to include the following additional steps in the generative process of LDA: 1) topic-specific tag-type distribution, 2) tag-type assignment for each word and 3) tag-type,topic-specific word distribution. The plate diagram for modified LDA is shown in Figure 7.

![Fig. 7: Modified LDA for handling semantic information.](Image)

We have computed Gibbs Sampling training equations for both extensions, and we plan to implement and test them soon.

E. Handling Multilinguality

Social media data related to climate change contains text from many different languages. It is important to develop similar algorithms and techniques for other languages to determine true global distributions of opinions. Hence, when we have satisfactory results for English, we plan to extend our research to other languages.

VII. CONCLUSION

In this report, we described the problem of sentiment analysis and its importance in the context of social media analysis. We also discussed a variety of approaches that are commonly used for the problem of sentiment analysis pertaining to three different paradigms of machine learning, i.e. unsupervised learning, supervised learning and ontology-based approaches. We provided a comparison of three approaches and discussed their advantages and disadvantages.

We also briefly mentioned our research plan in the coming future. Many of the ideas and algorithms discussed in the research plan are still part of ongoing work, and we plan to implement them soon and assess the results. We also plan to explore other machine learning paradigms such as deep learning for sentiment analysis [18] and statistical inference for knowledge-base construction [19].

REFERENCES

[16] nlp.stanford.edu/software/lex-parser.shtml