What Is an Opinion About? Exploring Political Standpoints Using Opinion Scoring Model

Bi Chen[1], **Leilei Zhu**[1], **Daniel Kifer**[2], **Dongwon Lee**[1]
[1]College of Information Sciences and Technology
[2]Department of Computer Science & Engineering
The Pennsylvania State University
bchen@ist.psu.edu, lleizhu@psu.edu, dan+aaai10@cse.psu.edu, dongwon@psu.edu

Abstract

In this paper, we propose a generative model to automatically discover the hidden associations between topics words and opinion words. By applying those discovered hidden associations, we construct the opinion scoring models to extract statements which best express opinionists' standpoints on certain topics. For experiments, we apply our model to the political area. First, we visualize the similarities and dissimilarities between Republican and Democratic senators with respect to various topics. Second, we compare the performance of the opinion scoring models with 14 kinds of methods to find the best ones. We find that sentences extracted by our opinion scoring models can effectively express opinionists' standpoints.

Introduction

"What do people think about ...?", this is a core question that opinion mining is trying to address. The question becomes more important as the web provides a ubiquitous platform for information exchange, where people show their personality and views. People record and share their feelings, express their like/dislike on products, give their voice to public issues, and so on. Opinion mining can help people better use those information and support their decision on diverse issues. For instance, the results of opinion mining in consumers' experiences and attitudes on different brands of cellphones will influence the consuming behavior of new customers.

In some of the early work on opinion mining focused on identifying the polarity of opinion words (Hatzivassiloglou and McKeown 1997), and document-level positive/negative sentiment classification, such as (Pang, Lee, and Vaithyanathan 2002). However, simply classifying documents as either positive or negative is not enough. Let's take a review on a camera as an example. A customer might like its lens system, and dislike its battery system. Researchers began to work on finer-grained opinion mining which mined opinions on different product features. This task is known as feature-level opinion mining, which could find, for instance, how customers evaluate a brand of camera's lens system or battery system, instead of the overall opinions.

Current opinion mining work mostly focuses on mining review data. There exist several reasons: 1) review data widely exists and are easy to obtain; 2) mining review data has their obvious business applications; 3) opinion words used in review normally have obvious sentiment orientations, such as good, bad and so on. However, if we extend opinion mining from the review domain to other domains, the situation makes more complex. For example, when a person talks about *iraq war*, someone might say "By removing Saddam Hussein, the world of the future is safer from terrorist attacks.", and others might say "The war will make people live in impoverished circumstances, and create civilian casualties." With regard to these statements, we cannot simply judge them to be either positive or negative.

When an opinionist express her opinion related to a certain topic, she will use some words more frequently than others. Continuing the above example, she will use words like *Saddam* and *war*, which tell people what topics she talks about. But these words are objective and cannot express her personal opinion. An opinionist will choose different words to express her opinion related to *iraq war* based on her stands. If one opinionist cares more about the safety situation, she will frequently use opinion words, like *safe*,*dangerous* and *attack*. If one opinionist cares more about the civilian situation, she will frequently use words, like *civilian*, *impoverished* and *injured*. From the example, we can see that although we cannot judge her opinion to be either positive or negative, we still can find associations between topic words and opinion words with regard to a certain opinion and topic. Such associations will help us to identify different stances among opinionists.

In this paper, topics are expressed through *noun* words, and opinions are conveyed through *adjective*, *verb* and *adverb* words. We propose a generative model to find associations between topic words and opinion words with regard to a certain opinionist and topic. In addition, we will construct a new opinion scoring model based on those found associations. By using this model, we will extract sentences which can represent an opinionist's stances on a certain topic from her statements. We will apply our proposed model to the political domain. By running our model on senators' statements, we will find similar/different stands of senators among two parties and corresponding sentences which mostly represents their stands.

Our contributions in this paper are: 1) propose a generative model to find hidden associations between topic and opinion words in an unsupervised way, 2) visualize opinionists' standpoints and identify controversial/consistent topics, and 3) build a series of opinion scoring models to extract statements which represent opinionists' stances. In the follows, we will discuss related work at first. Then, we will formulate our problem, and present the opinion scoring models in detail. In the final, we will do experiments on political standpoints visualization, and opinion sentence extraction.

Related Work

Opinion mining has been extensively studied in recent years. The most related work to ours is feature-level opinion mining. For a general survey, please refer to (Pang and Lee 2008). Our work is different from existed work in two main aspects: 1) Our proposed model identifies topics and associations between topics words and opinion words simultaneously , and 2) does not require topic (same with product feature) sets and opinion word sets to be manually specified in advance.

The early representative work is (Hu and Liu 2004) which uses association rule mining based method, and (Popescu and Etzioni 2005) which uses template extraction based method. Their methods explores associations between product features and opinion words by their explicit co-occurrence. Although they did a good job in identifying product features, they cannot detect features automatically. They identified product features by applying the synonym set in WordNet (Fellbaum 1998) and the semiautomated tagging of reviews. Our work finds topic sets (equal to product featues) automatically through topic models.

Topic-Sentiment Model (Mei et al. 2007) calculate sentiment coverage of documents by joint modeling the mixture of topics and sentiment predictions. But their model requires post-processing to calculate sentiment coverage of documents. Rather than post-processing, Joint Sentiment/Topic model (Lin and He 2009) can directly predict the sentiment orientation in the document level. Considering the hierarchy structure between objects and their associated aspects, Titov and McDonald (Titov and McDonald 2008b) proposed the Multi-Grain Latent Dirichlet Allocation model to find ratable aspects from global topics. Later, they proposed Multi-Aspect Sentiment model (Titov and McDonald 2008a) which summarizes sentiment texts by aggregating on each ratable aspects. However, in above work, researchers did not identify the associations between topics and sentiments. Our work identifies those associations automatically.

Above work does not identify hidden relations between feature groups and opinion words. (Takamura, Inui, and Okumura 2006) proposed a latent variable model to predict semantic orientation of phrases by finding associations between noun clusters and adjectives. However, their work does not cluster adjective words which leads to sparsity problem. (Su et al. 2008) and (Du and Tan 2009) clustered opinion word into groups and then found hidden associations between feature and opinion word groups by mutual reinforcement and information bottleneck algorithm respectively. However, their work need to predefine sets of words specifying positive and negative. We do not talk about positive and negative. Our goal is to extract opinions different from finding positive and negative sentences because we cannot easily use positive or negative criteria onto sentences in the field like politics.

Formal Statement

We start by providing a set of definitions that will be used in the remainder of this paper. In this paper, we will call opinion holder as an opinionist denoted by $a \in A$. Where, A is the set of all opinion holder. An opinionist can be a person, or a group who share similar opinions. A topic is a subject matter an opinionist talks about. In this paper, we define a topic $z \in Z$ as a multinomial distribution on noun words w^{noun}. An opinionist produces a collection of documents $\{D_1, D_2, ...D_i, ...D_n\}$, each of which expresses her opinions. Each document is a collection of statements $\{\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_i, ...\mathbf{w}_n\}$. In this paper, we choose each sentence is a statement. A statement $\mathbf{w}$ of an opinionist a is a set of words $\{w_1, w_2, ..., w_i, ...\}$, with i indicating the position in $\mathbf{w}$. The task of this paper is to build an opinion scoring model $Score(\mathbf{w}; a, z) = f(\{f_1(\mathbf{w}; a, z), f_2(\mathbf{w}; a, z), ..., f_i(\mathbf{w}; a, z), ..., f_n(\mathbf{w}; a, z)\})$ which assigns a real value to an opinionist a's statement $\mathbf{w}$ on a topic z, where $f_i(\mathbf{w}; a, z)$ represents i-th feature function and f is a map from a feature vector to a real value. If a statement $\mathbf{w}$ can better express her opinion on z, the opinion scoring model will assign a higher value to $\mathbf{w}$ than statements that cannot. By applying those feature functions f_i and the scoring function f, we will visualize opinionists' political standpoints, and find sentences that are the most representative of their opinion on a topic z.

Opinion Scoring Model

Model Overview

A statement $\mathbf{w}$ of an opinionist a could be either objective or subjective. His/her opinion is expressed through subjective ones. Even inside a subjective statement, objective and subjective information is mixed in an integrated and complex way. In order to score a statement $\mathbf{w}$ given an opinionist a and a topic z, we need to identify if it is subjective or objective. And if it is subjective, we also need to identify what topics she talks about as well as her opinion. Hence, we consider three kinds of features to score the opinion expressed by $\mathbf{w}$: subjective features, topic features and opinion features.

1. **Subjective features**. Subjective features captures whether a statement $\mathbf{w}$ expresses an opinion or not. A feature function $f_1(\mathbf{w})$ is defined on those features. If subjective features are found in a statement $\mathbf{w}$, $f_1(\mathbf{w})$ will return higher value than those statements without subjective features.
2. **Topic features**. Topic features identify what an opinionist talks about. Topics concerned in a statement $\mathbf{w}$ are expressed through noun words. A topic $z \in Z$ is defined as a multinomial distribution on noun words w^{noun}. $f_2(\mathbf{w}; \mathbf{z})$ is defined to capture topic features in $\mathbf{w}$. It will return a higher value if $\mathbf{w}^{noun}$ is more likely to be generated from a topic z.
3. **Opinion features**. Topics an opinionist talks about are conveyed by nouns, while opinions are expressed through adjective, verb and adverb words. If two opinionist have different opinions on a same topic, she will use different adjective, verb and adverb words to express their special opinions. Therefore, the usage patterns of adjective, verb

and adverb words are effective feature to capture an opinionist a's opinions on a topic z. We use three feature functions $f_3(\mathbf{w}^{adj}; a, z)$, $f_4(\mathbf{w}^{verb}; a, z)$ and $f_5(\mathbf{w}^{adv}; a, z)$ to capture the usage patterns of adjective, verb and adverb words respectively. $f_3(\mathbf{w}^{adj}; a, z)$ will return a higher value if $\mathbf{w}^{adj}$ is more likely to represent the usage of adjective words when a express her opinions on a topic z. $f_4(\mathbf{w}^{verb}; a, z)$ and $f_5(\mathbf{w}^{adv}; a, z)$ have same properties.

By incorporating above subjective, topic and opinion features, we can define the opinion scoring function as,

$$Score(\mathbf{w}; a, z) = f(f_1(\mathbf{w}), f_2(\mathbf{w}^{noun}; z), f_3(\mathbf{w}^{adj}; a, z), f_4(\mathbf{w}^{verb}; a, z), f_5(\mathbf{w}^{adv}; a, z)). \quad (1)$$

Obviously, Eq.1 is quite general, more feature functions can be included if needed. For convenience, we call $f_1(\mathbf{w})$ as the subjective function, $f_2(\mathbf{w}^{noun}; z)$ as the noun function, $f_3(\mathbf{w}^{adj}; a, z)$ as the adjective function, $f_4(\mathbf{w}^{verb}; a, z)$ as the verb function, $f_5(\mathbf{w}^{adv}; a, z)$ as the adverb function and f as the combination function. In the following we will discuss how to define them in detail.

Defining the Subjective Function

We choose *opinion clues* as basic criteria to judge whether a statement expresses an opinion or not. Or we could use OpinionFinder (Wilson et al. 2005) to label which sentences are subjective. *Opinion clues* are effective features used in (Furuse et al. 2007) to extract opinion sentences from blog pages. In this paper, we use rule-based method to define some *opinion clues*. Experiments show that rule-based clues are good enough for our application. It is also possible to collect more *opinion clues* though learning method as applied in (Riloff and Wiebe 2003). The following lists six clues we used. For more detail, please refer to (Furuse et al. 2007):

- Thought: think, consider, ...
- Impression: confuse, bewilder, ...
- Emotion: glad, worry, ...
- Modality about propositional attitude: should, would, ...
- Utterance-specific sentence form: however, nonetheless, ...
- Certainty/Uncertainty: wondering, questioning ...

In addition, we augment the above *opinion clues* by adding their synonyms through WordNet (Fellbaum 1998) and those opinion words included in MPQA, a corpus of opinion words (Wilson, Wiebe, and Hoffmann 2005).

The subjective feature function $f_1(\mathbf{w})$ is defined on the above *opinion clues*. If one or more *opinion clues* are found in a statement $\mathbf{w}$, the returned value is 1, otherwise 0. Notice that judging whether a statement $\mathbf{w}$ is sentiment or not is independent from a specific opinionist a or topic z. We have found that this simple subjective function works well for our purpose.

Noun Function

We use $p(\mathbf{w}^{noun}|z)$ to calculate $f_2(\mathbf{w}^{noun}; z)$. $p(\mathbf{w}^{noun}|z)$ is the probability of generating noun words in a statement $\mathbf{w}$ given a topic z. A widely used method is to treat $\mathbf{w}$ as a unigram model. We choose five different methods to calculate $f_2(\mathbf{w}^{noun}; z)$ from $p(w^{noun}|z)$. We use LDA model to calculate $p(w^{noun}|z)$. The only difference is that we use noun words to train the LDA model instead of all words. We run LDA on document level instead of statement level, which is too fine for LDA model. Through experiments, we find topics learned from noun words become more clear than topics learned from all words. Because of limited space, we do not introduce LDA model here, and please to refer to (Blei, Ng, and Jorda 2003) if interested.

1. **SumLog**. A simplest way is to choose the logarithm of the product of $p(w^{noun}|z)$. By considering the length of each statement, we divide the logarithm by the length of $\mathbf{w}^{noun}$.
   $f_2(\mathbf{w}^{noun}; z) = \sum_{w^{noun} \in \mathbf{w}^{noun}} \frac{1}{|\mathbf{w}^{noun}|} log(p(w^{noun}|z))$.
2. **SumBasic**. This algorithm is introduced from the **S**UM**B**ASIC (Nenkova and Vanderwende 2005), which is a simple effective sentence extraction algorithm for multi-document summarization.
   $f_2(\mathbf{w}^{noun}; z) = \sum_{w^{noun} \in \mathbf{w}^{noun}} \frac{1}{|\mathbf{w}^{noun}|} p(w^{noun}|z)$.
3. **Max@n(n=1,2,...)**. Instead of considering all noun words, we only consider n noun words $w^{noun} \in \mathbf{w}_n^{noun}$ which have higher values $p(w^{noun}|z)$ than the rest of noun words in a statement $\mathbf{w}$. In this paper, we will test Max@1, Max@2 and Max@3.
   $f_2(\mathbf{w}^{noun}; z) = \sum_{w^{noun} \in \mathbf{w}_n^{noun}} \frac{1}{n} p(w^{noun}|z)$.
4. **SimCos**. This algorithm treats $\mathbf{w}^{nouns}$ having an empirical unigram distribution $P_{\mathbf{w}^{noun}}$ on noun words. We use $cosine$ function to calculate the similarity between $P_{w^{noun}}$ and z.
   $f_2(\mathbf{w}^{noun}; z) = cosine(P_{w^{noun}}, z)$.
5. **SimKL**. Similar to **SimCos** , we use KL-Divergence to calculate the similarity between $P_{w^{noun}}$ and z. Considering f_2 has a higher value if $P_{w^{noun}}$ is close to z, we take the reciprocal form as,
   $f_2(\mathbf{w}^{noun}; z) = 1/KL(P_{w^{noun}}||z)$.

Adj/Verb/Adv Function

We still apply the same ideas used in **SumLog**, **SumBasic**, **Max@n**, **SimCos** and **SimKL** to calculate $f_3(\mathbf{w}^{adj}; a, z)$. Here, we only present how to calculate $f_3(\mathbf{w}^{adj}; a, z)$. The algorithm for calculating $f_4(\mathbf{w}^{verb}; a, z)$ and $f_5(\mathbf{w}^{adv}; a, z)$ is same. Similarly, we need to calculate $p(w^{adj}|a, z)$.

$p(w^{adj}|a, z)$ is trying to capture the usage pattern of adjective words when an opinionist a talks about topic z. For example, if an environmentalist talks on topics of energy, some adjective words, like *renewable*, *sustainable* and *clean* will be used more frequently than others. That is, $p(w^{adj}|a, z)$ is to discover relations between noun and adjective words. If we model their relations directly, we will face data sparsity problem. In order to reduce such a problem, we introduce a concept of *adjective class*, c^{adj}, to reduce the dimension of adjective words, like the concept *topic* used in LDA. Thus the question is changed to find relations between adjective classes c^{adj} and topics z.

We propose a generative model to learn the Adj function. We assume an opinionist a has a multinomial distribution

ψ_t on c^{adj} classes given a topic t. Given an opinionist a's statement $\mathbf{w}$, we have obtained its topic distribution θ after running LDA. Adjective words $w^{adj} \in \mathbf{w}$ are dependent on the topic distribution θ. The process of generating a adjective word w^{adj} is: first generate a topic z from θ, then generate an adjective class c^{adj} from ψ_t, and finally generate w^{adj} from c^{adj}. Formally, as illustrated in Fig. 1, the ADJ component assumes the following generative process for each adjective word w^{adj} in a statement $\mathbf{w}$ (with topic distribution θ) of an opinionist a:

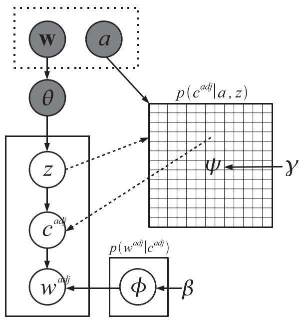

Figure 1: Generative Model for ADJ Component

Using Gibbs sampling techniques, we obtain following update equations for hidden variables c^{adj} and z on the i-th position as,

$$
\begin{aligned}
& p(c_i^{adj}|w_i^{adj}, z_i, \mathbf{c}_{-i}^{adj}, \mathbf{z}_{-i}) \\
= & \frac{N_{a,z,c^{adj}}(a, z_i, c_i^{adj}) + \gamma}{N_{a,z}(a, z_i) + |C^{adj}| \cdot \gamma} \cdot \frac{N_{c^{adj},w}(c_i^{adj}, w_i^{adj}) + \beta}{N_{c^{adj}}(c_i^{adj}) + |V^{adj}| \cdot \beta}, and \\
& p(z_i|c_i^{adj}, \mathbf{z}_{-i}, \mathbf{c}_{-i}^{adj}, \theta) \\
= & \frac{N_{a,z,c^{adj}}(a, z_i, c_i^{adj}) + \gamma}{N_{a,z}(a, z_i) + |C^{adj}| \cdot \gamma} \cdot p(z_i|\theta),
\end{aligned}
\tag{2}
$$

where $N_{a,z,c^{adj}}(a, z_i, c_i^{adj})$ is the number of adjective words belonging to opinionist a simultaneously assigned with adjective class c_i^{adj} and topic z_i; $N_{a,z}(a, z_i)$ is the integration of $N_{a,z,c^{adj}}(a, z_i, c_i^{adj})$ over adjective classes; $N_{c^{adj},w}(c_i^{adj}, w_i^{adj})$ is the number of adjective words w_i^{adj} assigned with adjective class c_i^{adj}; $N_{c^{adj}}(c_i^{adj})$ is the integration of $N_{c^{adj},w}(c_i^{adj}, w_i^{adj})$ on all adjective words; $|C^{adj}|$ is the number of adjective classes; and $|V^{adj}|$ is the size of adjective vocabulary.

From the model, we can learn $p(c^{adj}|a, z)$ and $p(w^{adj}|c^{adj})$. The Adj component $p(w^{adj}|a, z)$ can be obtained from $p(w^{adj}|a, z) = \sum_{c^{adj} \in C^{adj}} p(w^{adj}|c^{adj}) \cdot p(c^{adj}|a, z)$.

In essence, the relations between noun and adjective words we hope to discover are based on their co-occurrence. The boundary of co-occurrence in the current model is considered on statement level. If we use dependency parsing on statements in advance, we can reduce the boundary of co-occurrence, and find more accurate relations between noun and adjective words. We will leave it to future research.

Combination Function

We use two methods to combine above features. One is to train a linear regression model, as

$$
\begin{aligned}
f_{Linear} = & \alpha_0 + \alpha_1 \cdot f_1(\mathbf{w}) + \alpha_2 \cdot f_2(\mathbf{w}^{noun}; z) \\
& + \alpha_3 \cdot f_3(\mathbf{w}^{adj}; a, z) + \alpha_4 \cdot f_4(\mathbf{w}^{verb}; a, z) \\
& + \alpha_5 \cdot f_5(\mathbf{w}^{adv}; a, z)).
\end{aligned}
\tag{3}
$$

The other is to train a SVR model, as

$$
\begin{aligned}
f_{\mathrm{SVR}} = & \mathrm{SVR}(f_1(\mathbf{w}), f_2(\mathbf{w}^{noun}; z), f_3(\mathbf{w}^{adj}; a, z) \\
& f_4(\mathbf{w}^{verb}; a, z), f_5(\mathbf{w}^{adv}; a, z)).
\end{aligned}
\tag{4}
$$

We manually use some labeled data to learn a linear model and a SVR model. By incorporating **SumLog**, **SumBasic**, **Max@n**, **SimCos** and **SimKL** , we construct 10 opinion scoring model, annotated as **Linear-SumLog**, **Linear-SumBasic**, **Linear-Max@n**, **Linear-SimCos**, **Linear-SimKL**, **SVR-SumLog**, **SVR-SumBasic**, **SVR-Max@n**, **SVR-SimCos** and **SVR-SimKL** .

Experiments

Data Collection

We downloaded the statement records of senators through the Project Vote Smart WebSite [1]. These statement records present the political stances of senators. Because some senators retired and their records are not publicly available, we got a total $15,512$ statements from 88 senators. On average, each senator issued 176 statements of 214 words each. Then, we used the Part-of-Speech tagging function provided by MontyLingua Python library [2] to classify tokens into nouns, adjectives, verbs and adverbs. We total obtain $2,146,052$ noun words, $695,730$ adjective words, $412,468$ verb words, and $56,033$ adverb words. We also build a baseline where only subjectivity is considered.

Political Standpoints Visualization

Visualization of opinion can reduce users' cognitive efforts. Our opinion scoring model can be used for opinion visualization although it is not the main focus of our paper. In our first set of experiments, we use the associations identified by our model to visualize the similarities and dissimilarities between Republican and Democratic senators with respect to various topics.

We set the number of topics, Z, to be 200. We grouped adjectives, verbs, and adverbs into opinion word classes C^{opi}. Each topic was given 2 classes of opinion words (the idea is that one of the classes would be frequently associated with statements by Democrats and the other with statements by Republicans), so that the total number of opinion word classes C^{opi} is 400. Now, since some senators rarely make statements on certain issues, so for each of the discovered topics we examined the 20 senators who made the most statements about that topic. To quantify the difference between the Republican and Democratic stances on a topic z, we used the function $Diff(z)$ defined as:

$$
Diff(z) = |\frac{1}{|A^1|} \sum_{a \in A^1} (x_{z,1}^a - x_{z,2}^a) - \frac{1}{|A^2|} \sum_{a \in A^2} (x_{z,1}^a - x_{z,2}^a)| \tag{5}
$$

[1] http://www.votesmart.org

[2] http://web.media.mit.edu/hugo/montylingua/index.html

where a represents a senator, A^1 is the set of Democratic senators, and A^2 is the set of Republican senators. For each topic z and for each senator a, the quantities $x^a_{z,1}$ and $x^a_{z,2}$ are the components of the multinomial distribution (associated with senator a) over the two opinion classes associated with topic z. Due to space constraints, we only present 8 representative topics as well as how differences exist between two parties. The results are shown in Figure 2 (for readability, we manually labeled these 8 topics).

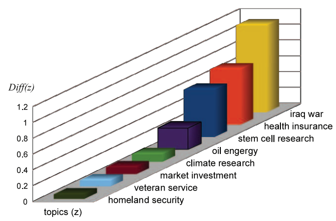

Figure 2: Different Stands Between Two Parties (For convenience, we only show human labeled topics instead of original distribution on noun words)

From Figure 2, we can see that the Democratic and Republic parties have quite different stances on topics of *Iraq war*, *health insurance* and *stem cell research*. On the other hand, two parties have quite similar stances on topics like *homeland security*, *veteran service*, *market investment* and *climate research*. With respect to the topic *oil energy*, two parties have mild differences.

We also manually checked the corresponding statements on these topics, and obtained the same results. For the topic of *Iraq war*, senators from the two parties hold entirely different views. Democrats think “The Iraq War has made America less secure and has been a major influence on our weakening economy. We owe it to all Americans to change course in Iraq and bring a responsible end to this war. (Harry Reid)”. They are criticized by the Republicans as “having given up on the idea of winning in Iraq (Lindsey Graham)”. *Stem Cell* research is another controversial battlefield. While the Democrats overwhelmingly praise it as “holding promise for the treatment of a number of diseases and conditions, and giving new hope for scientific breakthroughs (Bob Casey)”, the Republicans concern more on the ethicality issues. They emphasize that “Destroying viable embryos is not a choice we should or have to make.(Lindsey Graham)”. *Climate Change Research* gets support from both aisles. While John Kerry claims that “it’s about time we see the issue of global climate change receiving the attention it deserves.”, Olympia Snowe also states that “with science indicating a ninety-percent certainty of a direct link between human activity and climate change, Americans must take hold of this window of opportunity to reduce our current levels of carbon dioxide.”

This experiment shows that our model can effectively extract hidden associations between topic and opinion words for different opinionists. Those hidden associations also effectively represent opinionists’ stances on various topics. People are inclined to paying close attention to controversial topics. Our model provides a way to automatically discover those controversial public issues.

Opinion Sentence Extraction

Visualizing topics which are controversial or consistent between two parties is not enough. We also need to know their personal points of view. In this part, we will do a quantitative experiment to evaluate the performance of our proposed opinion scoring models. For the selected senator and topic, we will extract 5 sentences which can express their stands best using the opinion scoring model. Since our model is different from the models (Su et al. 2008; Du and Tan 2009) mentioned in the related section which need opinion word sets in advance. Both cannot be compared directly, and thus we only compare the models proposed in this paper. We instantiate the method **Max@n** to **Max@1** , **Max@2** and **Max@3** . So in total, we have 14 models for comparison.

We manually labeled $1,250$ sentences for training the combination model. We randomly selected 5 topics and selected one senator for each topic. For each combination of a topic and a senator, we extracted 250 sentences. We gave a score to each sentence based on the following criteria: 1) score 5: strong opinion sentence related to the given topic, 2) score 4: weak opinion sentence related to the given topic, 3) score 2: not an opinion sentence but related to the given topic, and 4) score 1: not an opinion sentence and not related to the given topic.

We select 15 topics, and 5 senators for each topic for testing. So we have 75 different combination of topics and senators. For each combination, we generate 5 sentences for each model. Thus we manually evaluate $5,250$ sentences. For the evaluation, we adopt three metrics, which capture the performance at different aspects:

- **Mean Reciprocal Rank (MRR)**. MRR measures the relevance of the first ranked sentence, averaged over all results. MRR provides the insight in the ability of the opinion scoring models to return a relevant sentence at the topic of the ranking.
- **Success at rank k (S@k)**. S@k defines the success at rank k, which reflects the probability of finding a relevant sentence among the top k recommended sentences. We will evaluate the results using S@1 and S@5.
- **precision at rank k (P@k)**. P@k reports the precision at rank k, which is defined as the proportion of extracted sentences that is relevant, averaged over all results. We will evaluate the results using P@5.

We have tested different settings for the number of topics, classes of adjective, verb and adverb words. When we set the topic number $Z = 200$, adjective class number $C^{adj} = 100$, verb class number $C^{verb} = 100$, and adverb class number $C^{adv} = 50$, we could obtain reasonable results for opinion sentence selection. Because of limited space, we only report results under those settings. Table 1 lists the results of opinion sentences extraction using 14 models.

From the Table 1, we can see the quite low precision of the baseline. Among all models, SVR non-linear method is the best. That means whether or not a sentence has strong/weak/non opinion associated with a topic is decided by a complex combination of its topic, adjective, verb and adverb features. With regard to different methods, we note that SVR-Max@1, SVR-Max@2 and SVR-SimCos obtain

Method	MRR	S@1	S@5	P@5
Linear-SumLog	0.69	0.51	0.61	0.39
SVR-SumLog	0.72	0.62	0.70	0.45
Linear-SumBasic	0.67	0.52	0.81	0.53
SVR-SumBasic	0.84	0.79	0.93	0.69
Linear-Max@1	0.93	**0.90**	**0.97**	0.83
SVR-Max@1	**0.95**	**0.90**	**0.97**	0.84
Linear-Max@2	0.82	0.75	**0.97**	0.69
SVR-Max@2	0.90	0.87	**0.97**	0.78
Linear-Max@3	0.79	0.65	0.90	0.61
SVR-Max@3	0.87	0.80	**0.97**	0.71
Linear-SimCos	0.91	0.85	**0.97**	0.81
SVR-SimCos	0.93	0.89	**0.97**	**0.85**
Linear-SimKL	0.79	0.72	0.82	0.73
SVR-SimKL	0.85	0.78	0.86	0.75
Baseline Model	<0.05	<0.05	<0.05	<0.05

Table 1: Results of Opinion Scoring Models

Feature Combination	MRR	S@1	S@5	P@5
Noun	0.50	0.47	0.58	0.38
Noun+Adj	0.90	0.84	0.93	0.80
Noun+Adj+Verb	**0.93**	**0.89**	**0.97**	**0.85**
Noun+Adj+Verb+Adv	**0.93**	**0.89**	**0.97**	**0.85**

Table 2: Contribution of Noun, Adjective, Verb and Adverb Features

the best performance. From this results, we can see the opinion and topic associated to a sentence is usually determined by one or two important words. Such a result is in accordance with our intuition. When we read a sentence, we can judge what it talks about and what opinion it expresses just using a few significant words, instead of the average words in that sentence. We also note that SVR-SimCos is better than SVR-SimKL. The reason is that Cosine is more prefer to high frequent components, while KL is more prefer to low frequent components.

Next, we examine how noun, adjective, verb and adverb features contribute to the opinion sentence extraction. We will quantify contributions of noun, adjective, verb and adverb features to the opinion sentence extraction under the SVR-SimCos model. (We obtain the same results under SVR-Max@1 and SVR-Max@2, and thus omit them).

The first row in the Table 2 is essentially a baseline where we only consider subjective and topic-related measures. The following rows show promotions after adjective, verb and adverb features applied. We can see that adjective feature are the most important feature for opinion sentence extraction. Verb features also contribute a little for opinion mining, but not as significant as adjective words. However, we do not see any contributions from adverb features. The first reason why adverb feature is not significant is that the number of adverb words are less than $1/7$ number of adjective and verb words. The associations between noun and adverb words are not clear as adjective and verb words do. Here, we give a concrete example, a topic of *Climate Change* and *California Senator Feinstein*, to show how adjective and verb features contribute for opinion sentences extraction. When he talked about this topic, he used adjective words such as *significant* and *environmental* with high frequency, and use verb words such as *combat* and *make* with high frequency. Hence, the model extracts opinion sentences, like "*Climate change is the most significant environmental challenge we face, and i believe that lowering the ethanol tariff will make it less expensive for the united states to combat global warming.*", to represent his opinion.

Conclusion and Future Work

In this paper, we build a generative model to find hidden associations between topics words and opinion words, and construct the opinion scoring models to extract sentences which can best represents opinionists' stances. In this paper, we do not use any grammar analysis among topic and opinion words. In the future work, we will apply grammar structure of sentences to help on identifying hidden associations between topics and opinion words, and promote the performance of the opinion scoring models.

References

Blei, D. M.; Ng, A. Y.; and Jorda, M. I. 2003. Latent dirichlet allocation. *In the Journal of Machine Learning Research*.

Du, W., and Tan, S. 2009. An interative reinforcement approach for fine-grained opinion mining. In *In NA-ACL'09*.

Fellbaum, C., ed. 1998. *WordNet: An Electronic Lexical Database*. Cambridge, MA: MIT Press.

Furuse, O.; Hiroshima, N.; Yamada, S.; and Kataoka, R. 2007. Opinion sentence search engine on open-domain blog. In *In IJCAI'07*.

Hatzivassiloglou, V., and McKeown, K. R. 1997. Predicting the semantic orientation of adjectives. In *In ACL'97*.

Hu, M., and Liu, B. 2004. Mining opinion features in customer reviews. In *In AAAI'04*, 755–760.

Lin, C., and He, Y. 2009. Joint sentiment/topic model for sentiment analysis. In *In CIKM'09*.

Mei, Q.; Ling, X.; Wondra, M.; Su, H.; and Zhai, C. 2007. Topic sentiment mixture: modeling facets and opinions in weblogs. In *In WWW'07*, 171–180.

Pang, B., and Lee, L. 2008. Opinion mining and sentiment analysis. In *Foundations and Trends in Information Retrieval*.

Pang, B.; Lee, L.; and Vaithyanathan, S. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *In EMNLP'02*.

Popescu, A., and Etzioni, O. 2005. Extracting product features and opinions from reviews. In *In HLT/EMNLP'05*.

Riloff, E., and Wiebe, J. 2003. Learning extraction patterns for subjective expressions. In *In EMNLP'03*.

Su, Q.; Xu, X.; Guo, H.; Guo, X.; Wu, X.; Xiaoxun Zhang, B. S.; and Su, Z. 2008. Hidden sentiment association in chinese web opinion mining. In *In WWW'08*.

Takamura, H.; Inui, T.; and Okumura, M. 2006. Latent variable models for semantic orientations of phrases. In *In EACL'06*.

Titov, I., and McDonald, R. 2008a. A joint model of text and aspect ratings for sentiment summarization. In *In ACL'08*.

Titov, I., and McDonald, R. 2008b. Modeling online reviews with multi-grain topic models. In *In WWW'08*.

Wilson, T.; Hoffmann, P.; Somasundaran, S.; Kessler, J.; Wiebe, J.; Choi, Y.; Cardie, C.; Riloff, E.; and Patwardhan, S. 2005. Opinionfinder: a system for subjectivity analysis. In *In HLT/EMNLP'05*.

Wilson, T.; Wiebe, J.; and Hoffmann, P. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *In HLT/EMNLP'05*.