Opinion Retrieval in Twitter

Zhunchen Luo
College of Computer
National University of Defense Technology
410073 Changsha, Hunan, CHINA
zhunchenluo@nudt.edu.cn

Miles Osborne
School of Informatics
The University of Edinburgh
EH8 9AB, Edinburgh, UK
miles@inf.ed.ac.uk

Ting Wang
College of Computer
National University of Defense Technology
410073 Changsha, Hunan, CHINA
tingwang@nudt.edu.cn

Abstract

We consider the problem of finding opinionated tweets about a given topic. We automatically construct opinionated lexica from sets of tweets matching specific patterns indicative of opinionated messages. When incorporated into a learning-to-rank approach, results show that this automatically opinionated information yields retrieval performance comparable with a manual method. Finally, topic-related specific structured tweet sets can help improve query-dependent opinion retrieval.

Introduction

Twitter is a popular online social networking service where people often share information or opinions about personalities, politicians or products. Most existing work on opinion in Twitter concentrates on analysing opinions expressed in tweets for a given topic. To the best of our knowledge there is no work on actually finding opinionated tweets. In this paper, we present the first study of opinion retrieval in Twitter. Relevant opinionated tweets should satisfy two criteria: (1) be relevant to the query and (2) contain opinions or comments about the query, irrespective of being positive or negative.

Search in Twitter can be harder than traditional search, largely due to tweets being often very short, and/or lacking in reliable grammatical style and quality. These factors reduce the effectiveness of opinion mining based upon traditional NLP techniques. Twitter also presents interesting opportunities for retrieval. The rich environment presents us with a myriad of social information over-and-above just using terms in a post (for example author information such as the number of posts) all of which potentially can improve on (opinion) retrieval performance. Additionally conventions have emerged in Twitter which structure tweets and this structuring can be a valuable hint when retrieving opinionated tweets. As an example, people usually add a comment before the convention "RT @usename" and many of these tweets are likely to be subjective. Importantly this structural information is topic independent.

In this paper, we use a standard machine learning approach to learn a ranking function for tweets that uses the available social features and opinionated feature in addition to traditional topic-relevant features such as the BM25 score. The experimental result shows that our ranking function is significantly better than a BM25 baseline for opinion retrieval (improving MAP by 60.22%). Additionally we propose a novel approach, using the social information and structural information of the tweets, to automatically generate a large number of accurate "pseudo" subjective tweets (PSTs) and "pseudo" objective tweets (POTs). These two tweet sets can be used as a corpus to derive lexicons for estimating the opinionatedness of a new tweet. We show that our approach can achieve comparable performance with a method which using manual tagged tweets corpus.

Related Work

Opinion retrieval in blogs and web documents has been studied in depth. Eguchi and Lavrenko (2006) firstly introduced opinion ranking formula which combine sentiment relevance models and topic relevance models into a generation model. This formula was shown to be effective on the MPQA corpus. Zhang and Ye (2008) and Huang and Croft (2009) also put forward their own way to unify sentiment relevance models and topic relevance models for ranking. Gerani, Carman and Crestani (2009) firstly investigated learning-to-rank for blog posts. All of this work is in the context of blogs or web documents, Twitter, however, is a novel domain and its rich social environment should be considered when modeling relevance.

In opinion retrieval estimating the opinionatedness of a document is essential. He et al. (2008) proposed an approach to calculate opinionatedness of a document based on subjective terms. These terms are automatically derived from manual tagged data. Jijkoun, de Rijke and Weerkamp (2010) present a method for automatically generating topic-specific subjective lexicons based on extract syntactic clues of manual tagged data. Unlike the work introduced above, Zhang, Yu and Meng (2007) used the reviews of some websites as a source of "pseudo" subjective sentences, and the Wikipedia documents as an external source of "pseudo" objective sentences. They assume that the subjective portion should be dominant in the reviews so that the effect of the objective portion can be neglected. The situation is the opposite in Wikipedia documents. They then used these datasets to build a SVM sentence classifier to estimate the opinionatedness of a document. An approach based on similar idea we also proposed in the context of Twitter, which using social information and structural information to automatically generate "pseudo" subjective tweets and "pseudo" subjective tweets, for opinion retrieval in Twitter.

Approach

Learning to Rank Framework

Learning to rank is a data driven approach which effectively incorporates a bag of features into the retrieval process. A bag of features, related to the relevance of a tweet, are extracted from tweets that have been labelled as being relevant. RankSVM (Joachims Thorsten, 1999) is used to train a ranking model.

Social Features

The following features capture useful aspects of Twitter and authors for opinion retrieval:

Twitter Specific Features. We use the following features related to the tweet itself: **Mention**, **URL**, **Hashtag**. In a tweet, people usually use "@" preceding a user name to reply other user (Mention). The text of this tweet is more likely to be "personal content". Previous work shows that "personal content" is on the whole more likely to contain opinions than "official content" (Gerani et al. 2011). Therefore, we use a binary feature indicating whether the tweet contains "@username" for tweet opinion retrieval. Sharing links in tweets is very popular in Twitter. Most tweets containing a link usually give an objective introduction to the links (e.g., tweets posted by the BBC News). Additionally, spam in Twitter often contains links. Hence, we use a feature indicating whether a tweet contains a link in our ranking model. A hashtag refers to a word in the tweet that begins with the "#" character. It is used to indicate the topic of the tweet. We use a binary feature whether the tweet contains a hashtag in our system.

Author Features. We use the following features related to the author of the tweet: **Statuses**, **Followers**, **Friends**, **Listed**. The number of tweets (statuses) the author has ever written is related to the activeness of the author. Intuitively, the most active authors are likely to be spammers who post very large number of tweets. Therefore, we use the number of statuses as a feature for tweets ranking. The number of followers indicates the popularity of the user. For example, the news media users usually have more followers than normal users. The number of friends also reflects the type of the user. For example, spammers often have large number of 'friends'. We develop these two features for tweets retrieval. A user can group their friends into different lists according to some criteria (e.g., the topic and social relationship). If a user is listed many times, it means that his tweets are interesting to a larger user population. We use a feature that measures how many times the author of a tweet has been listed for tweet ranking.

Opinionatedness Feature

Obviously estimating the opinionatedness score of a tweet is essential for opinion retrieval task. We adopt a lexicon-based approach, since it is simple and not dependent on machine learning techniques. However, a lexicon such as the MPQA Subjectivity Lexicon[1] which is widely used might not be effective in Twitter, since the textual content of tweet is often very short, and lacks reliable grammatical style and quality. Therefore, we use a corpus-derived lexicon to construct an opinion score for each tweet. We estimate the opinionatedness score of each tweet by calculating the average opinion score over certain terms. We use the chi-square value, based on manual tagged subjective tweets set and objective tweets set, to estimate the opinion score of a term. The score measures how dependent a term is with respect to the subjective tweets set or objective tweets set. For all terms in a tweet, we only keep the terms with a chi-square value no less than m when computing the opinion score. The estimated formula as follows:

	t	$\neg t$	Row total
Sub. set	O_{11}	O_{12}	O_{1*}
Obj. set	O_{21}	O_{22}	O_{2*}
Col. total	O_{*1}	O_{*2}	O

Table 1: Table for pearson's chi-square. $O_{1*} = O_{11} + O_{12}$; $O_{2*} = O_{21} + O_{22}$; $O_{*1} = O_{11} + O_{21}$; $O_{*2} = O_{12} + O_{22}$; $O = O_{11} + O_{12} + O_{21} + O_{22}$.

$$Opinion_{avg}(d) = \sum_{t \in d, \chi^2(t) \geq m} p(t|d) \cdot Opinion(t)$$

where $p(t|d) = c(t,d)/|d|$ is the relative frequency of a term t in tweet d. $c(t,d)$ is the frequency of term t in tweet d. $|d|$ is the number of terms in tweet d.

$$Opinion(t) = sgn(\frac{O_{11}}{O_{1*}} - \frac{O_{21}}{O_{2*}}) \cdot \chi^2(t)$$

where $sgn(*)$ is sign function. $\chi^2(t)$ calculates chi-square value of a term.

$$\chi^2(t) = \frac{(O_{11}O_{22} - O_{12}O_{21})^2 \cdot O}{O_{1*} \cdot O_{2*} \cdot O_{*1} \cdot O_{*2}}$$

O_{ij} in Table 1 is counted as the number of tweets having term t in the subjective/objective tweets set respectively. For example O_{12} is the number of tweets not having term t in the subjective tweets set.

Manually labelling the tweets necessary for constructing opinionated scoring is time-consuming and also topic-dependent. For example, tweets about "android" might contain opioninated terms "open", "fast" and "excellent", but these terms are unlikely to be the subjective clues of tweets related to some news event (e.g., "UK strike"). It is clearly impossible to tag a large number of tweets for every given topic. Therefore, we develop an approach to collect "pseudo" subjective tweets (PSTs) and "pseudo" objective tweets (POTs) automatically.

In Twitter, some simple structural information of tweets and users' information can be used to generate PSTs and

[1] http://www.cs.pitt.edu/mpqa/

POTs. For example people usually retweet another user's tweet and give a comment before this tweet. Tweets with this structure are more likely to be subjective. Many tweets posted by news agencies are likely to be objective tweets and these tweets usually contain links. We define these two types of tweets as follows:

1) **"Pseudo" Subjective Tweet (PST):** a tweet of the form "RT @username" with text before the retweet. For example, a tweet "*I thought we were isolated and no one would want to invest here! RT @BBCNews: Honda announces 500 new jobs in Swindon* `bbc.in/vT12YY`" is a Pseudo subjective tweet.
2) **"Pseudo" Objective Tweet (POT):** If a tweet satisfies two criteria: (1) it contains links and (2) the user of this tweet posted many tweets before and has many followers. This tweet is likely to be an objective tweet. E.g., "*#NorthKorea:#KimJongil died after suffering massive heart attack on train on Saturday, official news agency reports* `bbc.in/vzPGY5`".

Using the definition introduced above, it is easy for us to design patterns and collect a large number of PSTs and POTs from Twitter. We assume that the tweets in the PST set are all subjective tweets and the tweets in the POT set are all objective tweets. Although this is not 100% true, the subjective tweets portion should be dominant in the PST set so that the effect of the objective tweets portion can be neglected. It is opposite in the POT set. Since the structural information and authors' information are independent of the topic of a tweet, if there are a lot of tweets related to a given topic, it is easily to collect topic-dependent PSTs and POTs.

Experiments

Dataset and Experimental Settings

To the best of our knowledge, there is no annotated dataset for opinion retrieval in Twitter. Therefore, we created a new dataset for this task[2]. We crawled and indexed about 30 million tweets using the Twitter API in November 2011. All tweets are English. Using these tweets we implemented a search engine. Seven people (a woman and six men) were asked to use our search engine. They were allowed to post any query. Given a query the search engine would present a list of 100 tweets ranked based on the BM25 score. Based on the principle about the tweet whether expresses opinion about a given query, people assigned a binary label to every tweet. Finally we totally collected 50 queries and all judged tweets. The average query length was 1.94 words and the average number of relevant tweets per query was 16.62.

For learning to rank, SVM light[3] which implements the ranking algorithm is used. We use a linear kernel for training and report results for the best setting of parameters. In order to avoid overfitting the data we perform 10 fold cross-validation in our dataset. And we use *Mean Average Precision* (MAP) as the evaluation metric.

Results

We first investigate whether social features can improve opinion retrieval in Twitter. As a baseline, we use the ranking approach which uses the Okapi BM25 score of each tweet as a features for modeling. We combine each social feature with the **BM25** feature within our tweet ranking system. Table 2 shows the performance of each ranking model. We can see that using **Mention**, **URL**, **Statuses** and **Followers** features significant improves the results when used with the baseline (BM25) in isolation. It suggests some social information can indeed help opinion retrieval in Twitter. We see that the **URL** feature is the most effective feature, perhaps because most textual content in these tweets are objective introductions. Also, spammers usually post tweets including links and features dealing with links might help reduce spam. The effect of **URL**, **Statuses** and **Followers** features for tweets ranking also supports our approach of using social information and structural information to generate "pseudo" objective tweets. The improvement of ranking result using **Mention** feature supports the idea that "personal content" is on the whole more likely to contain opinions than "official content" (Gerani et al. 2011).

Next we investigate the opinionatedness feature for tweets ranking. To automatically generate PSTs and POTs, we design some simple patterns: For PSTs generation, we choose the tweets uses the convention "RT @username", with text before the first occurrence of this convention. Additionally we find that the length of the preceding text should be no less than 10 character. For POTs generation, we choose the tweets which contain a link, the author for each tweet has no less than 1,000 followers and has posted at least 10,000 tweets. In our one month tweets dataset, 4.64% tweets are high quality PSTs and 1.35% tweets are POTs.

We spot-checked the quality of our automatically harvested tweets by randomly selected 100 PSTs and 100 POTs and manually inspecting them, judging the extent to which there were subjective or objective. In these tweets, 95% PSTs were subjective tweets and 85% POTs were objective tweets. This supports the idea that our approach can generate a large number of accurate PSTs and POTs. Hence, we randomly choose 4500 English PSTs and POTs to form a topic-independent dataset.

Another advantage of our approach is that it is easy to gather topic-dependent PSTs and POTs. We use all PSTs and POTs introduced above to implement a search engine. Given a query, the search engine can give any number of query-dependent PSTs and POTs ranked by BM25 score. We generate 4500 query-dependent PSTs and POTs for each query. In our corpus-derived approach, we use the Porter English stemmer and stop words to preprocess the text of tweets. Using these tweet datasets we can calculate the value of opinionatedness score for a new tweets. To achieve the best performance of tweets ranking, we set the threshold of m is 5.02 corresponding to the significance level of 0.025 for each term in dataset. This setting is the same as Zhang, Yu and Meng (2007)'s work. We call the feature using topic-independent dataset to estimate the opinionatedness score as **Q_I** feature and using topic-dependent datasets as **Q_D** feature. Previous work uses manual tagged tweets to esti-

[2] This dataset is available at `https://sourceforge.net/projects/ortwitter/`

[3] `http://svmlight.joachims.org/`

	MAP		MAP
BM25	0.2509	BM25+Statuses	0.2726$^{\triangle}$
BM25+Mention	0.2814$^{\triangle}$	BM25+Followers	0.2532$^{\triangle}$
BM25+URL	0.3380$^{\blacktriangle}$	BM25+Friends	0.2454
BM25+Hashtag	0.2384	BM25+Listed	0.2510
BM25+Q_I	0.3602$^{\blacktriangle}$	BM25+Gold	0.3615$^{\blacktriangle}$
BM25+Q_D	0.3667$^{\blacktriangle}$	Best	0.4020$^{\blacktriangle}$

Table 2: Performance of Ranking Method. A significant improvement over the BM25 ranking method with $^{\triangle}$ and $^{\blacktriangle}$ (for $p < 0.05$ and $p < 0.01$).

mate the opinionatedness score of a new tweet. In our experiment we use training data in each fold as the manual tagged tweets. We compare the method, using **Gold** feature based on these manual tagged tweets, with the method using our **Q_I** feature and **Q_D** feature for tweet ranking.

Table 2 shows the result of ranking using the opinionatedness features. We can see that all the methods, using opinionatedness features, improve the opinion retrieval performance over the BM25 method. It shows estimating the opinionatedness score of the tweet is essential for opinion retrieval task. The ranking method using **Q_I** feature or **Q_D** feature can achieve comparable performance with the BM25+Gold method (there are no significant difference at p=0.05). It suggests that using social information and structural information to generate accurate PSTs and POTs automatically is useful for opinion retrieval in Twitter. Importantly this method does not need any manually tagged tweets. We can also see that BM25+Q_D ranking method significantly improves the opinion retrieval over the BM25+ Q_I ranking method (at p=0.05). It means our approach can help resolving query-dependent problem.

Table 3 shows some opinion terms derived from different PST and POT sets. We can see that our approach can assign high scores to terms such as personal pronoun (e.g., "i", "u"and "my") and emotions (e.g., ":)", ":(" and ":d"). The reason is that personal content tweets are more likely to be subjective tweets. And for query-dependent PSTs and POTs, our approach successfully extract the opinionated feature "excit" ($Opinion(t) > 0$) which can express attitude about the movie "Breaking Dawn", and this term is unlikely to be used in the opinionated tweets related "UK strike" topic. In PSTs and POTs related to the "UK strike" topic, we discover (unsurprisingly) that the term "bbc" ($Opinion(t) < 0$) is more likely to appear in the objective tweets posted by BBC news.

Finally we add all the features which can significantly improve the opinion retrieval in Twitter into a ranking model. They are **BM25**, **Mention**, **URL**, **Statuses**, **Followers** and **Q_D** features. Table 2 shows the best result of method which improves MAP by 60.22% over the BM25 ranking method.

Conclusion

To the best of our knowledge, we are the first to propose a ranking model for opinion retrieval in Twitter. This model integrates social and opinionatedness information for tweets opinion retrieval. The experimental result shows that opinion retrieval performance is improved when links, mentions, author information such as the number of statues or followers and the opinionatedness of the tweet are taken into account. We also proposed a novel automatic approach which uses the social information and structural information of the tweets to generate accurate "pseudo" subjective tweets (PSTs) and "pseudo" subjective tweets (POTs) automatically. Opinionated retrieval results using this information is comparable to results using manually labelled data.

Q_I	Sub	i, lol, .., :), *, u, my, :d, me, morn
	Obj	new, via, ..., video, tip, social, 2011
Breaking Dawn	Sub	i, go, me, lol, !!!, excit, im, :), so, too
	Obj	video, premier, kristen, robert,
UK strike	Sub	i, you, my, lol, :(, u, me, so, !!, good
	Obj	followfridai, week, bbc, #ows, #jobs

Table 3: Opinion Terms Derived from Query-Independent PSTs and POTs (Q_I) and Query-Dependent PSTs and POTs (Breaking Dawn and UK strike) Respectively. "Sub" ("Obj") is the type of the terms which the value of their $Opinion(t)$ score are more (less) than 0.

Acknowledgements

We would like to thank Victor Lavrenko for his comments and Bo Dai, Ming Lei, Zhengshuai Lin, Xianling Mao, Hongguang Ren, He Wang, Chenkun Wu and Xianglilan Zhang for tagging the data. The research is supported by the National Natural Science Foundation of China (Grant No. 61170156 and 60933005) and a CSC scholarship.

References

Eguchi, K., and Lavrenko, V. 2006. Sentiment retrieval using generative models. In *EMNLP*.

Gerani, S.; Carman, M. J.; and Crestani, F. 2009. Investigating Learning Approaches for Blog Post Opinion Retrieval. In *ECIR*.

Gerani, S.; Keikha, M.; Carman, M.; and Crestani, F. 2011. Personal Blog Retrieval Using Opinion Features. In *ECIR*.

He, B.; Macdonald, C.; He, J.; and Ounis, I. 2008. An effective statistical approach to blog post retrieval. In *CIKM*.

Huang, X., and Croft, W. B. 2009. A unified relevance model for opinion retrieval. In *CIKM*.

Jijkoun, V.; de Rijke, M.; and Weerkamp, W. 2010. Generating focused topic-specific sentiment lexicons. In *ACL*.

Thorsten, J. 1999. Making Large-Scale SVM Learning Practical. In *Advances in Kernel Methods: Support Vector Learning*.

Zhang, M., and Ye, X. 2008. A generation model to unify topic relevance and lexicon-based sentiment for opinion retrieval. In *SIGIR*.

Zhang, W.; Yu, C.; and Meng, W. 2007. Opinion retrieval from blogs. In *CIKM*.