IMASS: An Intelligent Microblog Analysis and Summarization System

Jui-Yu Weng Cheng-Lun Yang Bo-Nian Chen Yen-Kai Wang Shou-De Lin
Department of Computer Science and Information Engineering
National Taiwan University
{r98922060,r99944042,f92025,b97081,sdlin}@csie.ntu.edu.tw

Abstract

This paper presents a system to summarize a Microblog post and its responses with the goal to provide readers a more constructive and concise set of information for efficient digestion. We introduce a novel two-phase summarization scheme. In the first phase, the post plus its responses are classified into four categories based on the intention, interrogation, sharing, discussion and chat. For each type of post, in the second phase, we exploit different strategies, including opinion analysis, response pair identification, and response relevancy detection, to summarize and highlight critical information to display. This system provides an alternative thinking about machine-summarization: by utilizing AI approaches, computers are capable of constructing deeper and more user-friendly abstraction.

1 Introduction

As Microblog services such as Twitter have become increasingly popular, it is critical to reconsider the applicability of the existing NLP technologies on this new media sources. Take summarization for example, a Microblog user usually has to browse through tens or even hundreds of posts together with their responses daily, therefore it can be beneficial if there is an intelligent tool assisting summarizing those information.

Automatic text summarization (ATS) has been investigated for over fifty years, but the majority of the existing techniques might not be appropriate for Microblog write-ups. For instance, a popular kind of approaches for summarization tries to identify a subset of information, usually in sentence form, from longer pieces of writings as summary (Das and Martins, 2007). Such extraction-based methods can hardly be applied to Microblog texts because many posts/responses contain only one sentence.

Below we first describe some special characteristics that deviates the Microblog summarization task from general text summarization.

a. The number of sentences is limited, and sentences are usually too short and casual to contain sufficient structural information or cue phrases. Unlike normal blogs, there is a strict limitation on the number of characters for each post (e.g. 140 characters for Twitter and Plurk maximum). Microblog messages cannot be treated as complete documents so that we cannot take advantage of the structural information. Furthermore, users tend to regard Microblog as a chatting board. They write casually with slangs, jargons, and incorrect grammar.
b. Microblog posts can serve several different purposes. At least three different types of posts are observed in Microblogs, expressing feeling, sharing information, and asking questions. Structured language is not the only means to achieve those goals. For example, people sometimes use attachment, as links or files, for sharing, and utilize emoticons and pre-defined qualifiers to express their feelings. The diversity of content differ Microblogs from general news articles. Consequently, using one mold to fit all types of Microblog posts is not sufficient. Different summarization schemes for posts with different purposes are preferred.
c. Posts and responses in Microblogs are more similar to a multi-persons dialogue corpus. One of the main purposes of a Microblog is to serve as the fast but not instant communication channel among multiple users. Due to the free-chatting, multi-user characteristics, the topic of a post/response thread can drift quickly. Sometimes, the topic of discussion at the end of the thread is totally unrelated to that of the post.

Proceedings of the ACL-HLT 2011 System Demonstrations, pages 133–138,
Portland, Oregon, USA, 21 June 2011.

This paper introduces a framework that summarizes a post with its responses. Motivated by the abovementioned characteristics of Microblogs, we plan to use a two-phase summarization scheme to develop different summarization strategies for different type of posts (see Figure 1). In the first phase, a post will be automatically classified into several categories including interrogation, discussion, sharing and chat based on the intention of the users. In the second phase, the system chooses different summarization components for different types of posts.

The novelties of this system are listed below.

1. Strategically, we propose an underlying 2-phase framework for summarizing Microblog posts. The system can be accessed online at http://mslab.csie.ntu.edu.tw/~fishyz/plurk/.
2. Tactically, we argue that it is possible to integrate post-intention classification, opinion analysis, response relevancy and response-pair mining to create an intelligent summarization framework for Microblog posts and responses. We also found that the content features are not as useful as the temporal or positional features for text mining in Microblog.
3. Our work provides an alternative thinking about ATS. It is possible to go beyond the literal meaning of summarization to exploit advanced text mining methods to improve the quality and usability of a summarization system.

2 Summarization Framework and Experiments

Below we discuss our two-phase summarization framework and the experiment results on each individual component. Note that our experiments were tested on the Plurk dataset, which is one of the most popular micro-blogging platforms in Asia.

Our observation is that Microblog posts can have different purposes. We divide them into four categories, *Interrogation*, *Sharing*, *Discussion*, and *Chat*.

The *Interrogation* posts are questions asked in public with the hope to obtain some useful answers from friends or other users. However, it is very common that some repliers do not provide meaningful answers. The responses might serve the purpose for clarification or, even worse, have nothing to do with the question. Hence we believe the most appropriate summarization process for this kind of posts is to find out which replies really respond to the question. We created a *response relevance detection* component to serve as its summarization mechanism.

The *Sharing* posts are very frequently observed in Microblog as Microbloggers like to share interesting websites, pictures, and videos with their friends. Other people usually write down their comments or feelings on the shared subjects in the responses. To summarize such posts, we obtain the statistics on how many people have positive, neutral, and negative attitude toward the shared subjects. We introduce the *opinion analysis* component that provides the analysis on whether the information shared is recommended by the respondents.

We also observe that some posts contain characteristics of both *Interrogation* and *Sharing*. The users may share a hyperlink and ask for others' opinions at the same time. We create a category named *Discussion* for these posts, and apply both response ranking and opinion analysis engines on this type of posts.

Finally, there are posts which simply act as the solicitation for further chat. For example, one user writes "so sad…" and another replies "what happened". We name this type of posts/responses as *Chat*. This kind of posts can sometimes involve multiple persons and the topic may gradually drift to a different one. We believe the plausible summarization strategy is to group different messages based on their topics. Therefore for *Chat* posts, we designed a *response pair identification* system to accomplish such goal. We group the related responses together for display, and the number of groups represents the number of different topics in this thread.

Figure 1 shows the flow of our summarization

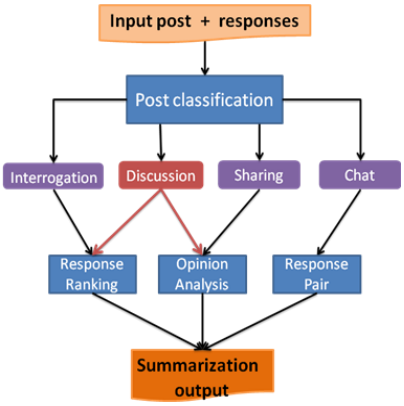

Figure 1. System architecture

framework. When an input post with responses comes in, the system first determines its intention, based on which the system adopts proper strategies for summarization. Below we discuss the technical parts of each sub-system with experiment results.

2.1 Post Intention Classification

This stage aims to classify each post into four categories, *Interrogation*, *Sharing*, *Discussion*, and *Chat*. One tricky issue is that the *Discussion* label is essentially a combination of interrogation and sharing labels. Therefore, simply treating it as an independent label and use a typical multi-label learning method can hurt the performance. We obtain 76.7% (10-fold cross validation) in accuracy by training a four-class classifier using the 6-gram character language model. To improve the performance, we design a decision-tree based framework that utilizes both manually-designed rules and discriminant classification engine (see Figure 2). The system first checks whether the posts contains URLs or pointers to files, then uses a binary classifier to determine whether the post is interrogative.

For the experiment, we manually annotate 6000 posts consisting of 1840 *interrogation*, 2002 *sharing*, 1905 *chat*, and 254 *discussion* posts. We train a 6-gram language model as the binary interrogation classifier. Then we integrate the classifier into our system and test on 6000 posts to obtain a testing accuracy of 82.8%, which is significantly better than 76.7% with multi-class classification.

2.2 Opinion Analysis

Opinion analysis is used to evaluate public preference on the shared subject. The system classifies responses into 3 categories, positive, negative, and neutral.

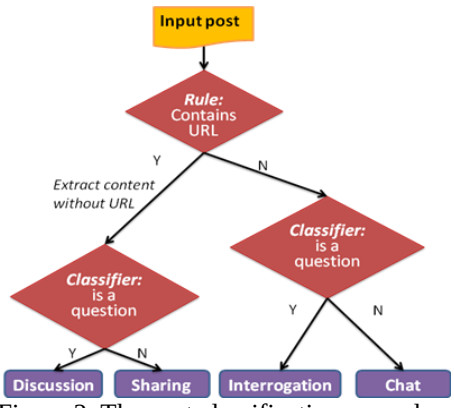

Figure 2. The post classification procedure

Here we design a two-level classification framework using Naïve-Bayes classifiers which takes advantage of the learned 6-gram language model probabilities as features. First of all, we train a binary classifier to determine if a post or a reply is opinionative. This step is called the subjectivity test. If the answer is yes, we then use another binary classifier to decide if the opinion is positive or negative. The second step is called the polarity test.

For subjectivity test, we manually annotate 3244 posts, in which half is subjective and half is objective. The 10-fold cross validation shows average accuracy of 70.5%.

For polarity test, we exploit the built-in emoticons in Plurk to automatically extract posts with positive and negative opinions. We collect 10,000 positive and 10,000 negative posts as training data to train a language model of Naïve Bayes classifier, and evaluate on manually annotated data of 3121 posts, with 1624 positive and 1497 negative to obtain accuracy of 0.722.

2.3 Response Pair Identification

Conversation in micro-blogs tends to diverge into multiple topics as the number of responses grows. Sometimes such divergence may result in responses that are irrelevant to the original post, thus creating problems for summarization. Furthermore, because the messages are usually short, it is difficult to identify the main topics of these dialogue-like responses using only keywords in the content for summarization. Alternatively, we introduce a subcomponent to identify Response Pairs in micro-blogs. A Response Pair is a pair of responses that the latter specifically responds to the former. Based on those pairs we can then form clusters of messages to indicate different group of topics and mes-

Feature	*Description*	*Weight*
Backward Referencing	Latter response content contains former responder's display name	0.055
Forward Referencing of user name	Former response contains latter response's author's user name	0.018
Response position difference	Number of responses in between responses	**0.13**
Content similarity	Contents' cosine similarity using n-gram models.	0.025
Response time difference	Time difference between responses in seconds	0.012

Table 1. Feature set with their description and weights

sages.

Looking at the content of micro-blogs, we observe that related responses are usually adjacent to each other as users tend to closely follow whether their messages are responded and reply to the responses from others quickly. Therefore besides content features, we decide to add the temporal and ordering features (See Table 1) to train a classifier that takes a pair of messages as inputs and return whether they are related. By identifying the response pairs, our summarization system is able to group the responses into different topic clusters and display the clusters separately. We believe such functionality can assist users to digest long Microblog discussions.

For experiment, the model is trained using LIBSVM (Chang and Lin, 2001) (RBF kernel) with 6000 response pairs, half of the training set positive and half negative. The positive data can be obtained automatically based on Plurk's built in annotation feature. Responses with @user_name string in the content are matched with earlier responses by the author, user_name. Based on the learned weights of the features, we observe that content feature is not very useful in determining the response pairs. In a Microblog dialogue, respondents usually do not repeat the question nor duplicate the keywords. We also have noticed that there is high correlation between the responses relatedness and the number of other responses between them. For example, users are less likely to respond to a response if there have been many replies about this response already. Statistical analysis on positive training data shows that the average number of responses between related responses is 2.3.

We train the classifier using 6000 automatically-extracted pairs of both positive and negative instances. We manually annotated 1600 pairs of data for testing. The experiment result reaches 80.52% accuracy in identifying response pairs. The baseline model which uses only content similarity feature reaches only 45% in accuracy.

2.4 Response Relevance Detection

For interrogative posts, we think the best summary is to find out the relevent responses as potential answers.

We introduce a *response relevancy detection* component for the problem. Similar to previous components, we exploit a supervised learning approach and the features' weights, learned by LIBSVM with RBF kernel, are shown in Table 2.

Feature	*Weight*
Response position	**0.170**
Response time difference	0.008
Response length	0.003
Occurrence of interrogative words	0.023
Content similarity	0.023

Table 2. Feature set and their weights

Temporal and Positional Features

A common assertion is that the earlier responses have higher probability to be the answers of the question. Based on the learned weights, it is not surprising that most important feature is the position of the response in the response hierarchy. Another interesting finding by our system is that meaningful replies do not come right away. Responses posted within ten seconds are usually for chatting/clarification or ads from robots.

Content Features

We use the length of the message, the cosine similarity of the post and the responses, and the occurrence of the interrogative words in response sentences as content features.

Because the interrogation posts in Plurk are relatively few, we manually find a total of 382 positive and 403 negative pairs for training and use 10-fold cross validation for evaluation.

We implement the component using LIBSVM (RBF Kernel) classifier. The baseline is to always select the first response as the only relevant answer. The results show that the accuracy of baseline reaches 67.4%, far beyond that of our system 73.5%.

3 System Demonstration

In this section, we show some snapshots of our summarization system with real examples using Plurk dataset. Our demo system is designed as a

Figure 3. The IMASS interface

Original post and responses:

iantearle	**asks**	**if anyone can send him the Helvetica font for mac???!!! PLEASSSSSEEEEE**
trinitydechou	has	it, but not at mac.... so if you don't get it later, let me know I can fire it over to you.
iantearle	will	let you know!
trinitydechou	:	bounces on you
iantearle	:	naughty!
iantearle	:	i do
iantearle	:	Ian check your email, i sent you the fonts

Summarization:

iantearle	**asks**	**if anyone can send him the Helvetica font for mac???!!! PLEASSSSSEEEEE**
trinitydechou	has	it, but not at mac.... so if you don't get it later, let me know I can fire it over to you.
iantearle	:	Ian check your email, i sent you the fonts

Figure 4. An example of interrogative post.

search engine (see Figure 3). Given a query term, our system first returns several posts containing the query string under the search bar. When one of the posts is selected, it will generate a summary according to the detected intention and show it in a pop-up frame. We have recorded a video demonstrating our system. The video can be viewed at http://imss-acl11-demo.co.cc/.

For interrogative posts, we perform the response relevancy detection. The summary contains the question and relevant answers. Figure 4 is an example of summary of an interrogative post. We can see that responses other than the first and the last responses are filtered because they are less relevant to the question.

For sharing posts, the summary consists of two parts. A pie chart that states the percentage of each opinion group is displayed. Then the system picks three responses from the majority group or one response from each group if there is no significant difference. Figure 5 is an example that most friends of the user *dfrag* give positive feedback to the shared video link.

For discussion posts, we combine the response relevancy detection subsystem and the opinion analysis sub-system for summarization. The former first eliminates the responses that are not likely to be the answer of the post. The latter then generates a summary for the post and relevant responses. The result is similar to sharing posts.

For chat posts, we apply the response pair identification component to generate the summary. In the example, Figure 6, the original Plurk post is about one topic while the responses diverge to one

Original post and responses:

dfrag	**:**	**http://www.youtube.com/watch?v=KgISPI7g14Q Yo, fo' real. Boo yaakasha!**
CatC	says	mornin'
dfrag	:	That yawning smiley made me yawn! WTF!?
CatC	says	lol
CatC	says	my work here is done.
dfrag	:	LoL
TVisio	:	Who do you feel more sorry for? One is an anal retentive curmudgeon, the other is an anal Yo!
Snailrind	loves	Sacha Baron Cohen.
Snailrind	thinks	he's at his best when his interviewee has no idea who he is.
1bzymama	:	Meh. It was alright
Snailrind	says	"is it because I is black?"

Summarization:

Recommend!!

dfrag	**:**	**http://www.youtube.com /watch?v=KgISPI7g14Q Yo, fo' real. Boo yaakasha!**	***negative***
CatC	says	lol	*positive*
CatC	says	my work here is done.	*positive*
dfrag	:	LoL	*positive*

Figure 5. An example of sharing post.

Original post and responses:

TwilaMarie	**:**	**time to catch up on Dollhouse. lovin tv-dome.net!**
chaotixfusion	:	
TwilaMarie	:	hiya Rio! Git your message this evening
TwilaMarie	:	got8 LOL
rthefish	:	Having a hard time getting up in that show. I have the past 2 I need to watch though.
TwilaMarie	:	i really like it! love all the free pc-tv places to catch up on things. spent the last two days doing just that
chaotixfusion	:	ahh okay , at least you got it

Summarization:

TwilaMarie(*TwilaMarie*): time to catch up on Dollhouse. lovin tv-dome.net!
Rio(*chaotixfusion*):
TwilaMarie(*TwilaMarie*): hiya Rio! Git your message this evening
TwilaMarie(*TwilaMarie*): got8 LOL
Rio(*chaotixfusion*): ahh okay , at least you got it

TwilaMarie(*TwilaMarie*): time to catch up on Dollhouse. lovin tv-dome.net!
rthefish(*rthefish*): Having a hard time getting up in that show. I have the past 2 I need to watch though.
TwilaMarie(*TwilaMarie*): i really like it! love all the free pc-tv places to catch up on things. spent the last two days doing just that

Figure 6. An Example of chat post

or more unrelated topics. Our system clearly separates the responses into multiple groups. This representation helps the users to quickly catch up with the discussion flow. The users no longer have to read interleaving responses from different topics and guess which topic group a response is referring to.

4 Related Work

We have not seen many researches focusing on the issues of Microblog summarization. We found only one work that discusses about the issues of summarization for Microblogs (Sharifi et al., 2010). Their goal, however, is very different from ours as they try to summarize multiple posts and do not consider the responses. They propose the Phrase Reinforcement Algorithm to find the most commonly used phrase that encompasses the topic phrase, and use these phrases to compose the summary. They are essentially trying to solve a multi-document summarization problem while our problem is more similar to short dialog summarization because the dialogue nature of Microblogs is one of the most challenging part that we tried to overcome.

In dialogue summarization, many researchers have pointed out the importance of detecting response pairs in a conversation. Zechner (2001) performs an in depth analysis and evaluation in the area of open domain spoken dialogue summarization. He uses decision tree classifier with lexical features like POS tags to identify questions and applies heuristic rules like maximum distance between speakers to extract answers. Shrestha and McKeown (2004) propose a supervised learning method to detect question-answer pairs in Email conversations. Zhou and Hovy (2005) concentrates on summarizing dialogue-style technical internet relay chats using supervised learning methods. Zhou further clusters chat logs into several topics and then extract some essential response pairs to form summaries. Liu et al. (2006) propose to identify question paragraph via analyzing each participant's status, and then use cosine measure to select answer paragraphs for online news dataset.

The major differences between our components and the systems proposed by others lie in the selection of features. Due to the intrinsic difference between the writing styles of Microblog and other online sources, our experiments show that the content feature is not as useful as the position and temporal features.

5 Conclusion

In terms of length and writing styles, Microblogs possess very different characteristics than other online information sources such as web blogs and news articles. It is therefore not surprising that different strategies are needed to process Microblog messages. Our system uses an effective strategy to summarize the post/response by first determine the intention and then perform different analysis depending on the post types. Conceptually, this work intends to convey an alternative thinking about machine-summarization. By utilizing text mining and analysis techniques, computers are capable of providing more intelligent summarization than information condensation.

Acknowledgements

This work was supported by National Science Council, National Taiwan University and Intel Corporation under Grants NSC99-2911-I-002-001, 99R70600, and 10R80800.

References

Chih-Chung Chang and Chih-Jen Lin. 2001. LIBSVM : a library for support vector machines. Software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm

Dipanjan Das and André F.T. Martins. 2007. A Survey on Automatic Text Summarization. Literature Survey for the Language and Statistics II Course. CMU.

Chuanhan Liu, Yongcheng Wang, and Fei Zheng. 2006. Automatic Text Summarization for Dialogue Style. In Proceedings of the IEEE International Conference on Information Acquisition. 274-278

Beaux Sharifi, Mark A. Hutton, and Jugal Kalita. 2010. Summarizing Microblogs Automatically. In Proceedings of the Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT). 685-688

Lokesh Shrestha and Kathleen McKeown. 2004. Detection of Question-Answer Pairs in Email Conversations. In Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010).

Klaus Zechner. 2001. Automatic Generation of Concise Summaries of Spoken Dialogues in Unrestricted Domains. In Proceedings of the 24th ACM-SIGIR International Conference on Research and Development in Information Retrieval. 199-207.

Liang Zhou and Eduard Hovy. 2005. Digesting virtual geek culture: The summarization of technical internet relay chats, in Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL 2005). 298-305.