Topic Detection in Instant Messages

Han Zhang*, Chang-Dong Wang* and Jian-Huang Lai†
*School of Mobile Information Engineering
Sun Yat-sen University, Zhuhai, P. R. China, 519082
Email: zh950713@gmail.com, changdongwang@hotmail.com
†School of Information Science and Technology
Sun Yat-sen University, Guangzhou, P. R. China, 510006
SYSU-CMU Shunde International Joint Research Institute (JRI), Shunde, P.R. China, 528300
Email:stsljh@mail.sysu.edu.cn

Abstract—In the past few years, instant messaging (IM) has been widely used in daily communication. However, due to the dispersion of topics and meaningless chatting, online IM groups are filled with useless messages. In order to help IM users capture what the IM group is talking about without reading all the messages, topic discovery in instant messages becomes a significant but challenging research task. In this paper, we propose a new method for topic detection in instant messages, which is applicable for the case where 1) useless terms keep emerging; 2) the instant messages are very short; and 3) multiple languages are used. The basic step is to treat each message in an online group discussion as a data item in message stream, and then apply PLSA on the collected instant messages. One strategy is designed to segment multilingual message without utilizing machine translation and remove the useless words that keep emerging. Extensive experiments conducted on the real-world QQ group data confirm the effectiveness of the proposed method.

Keywords—Topic detection; Instant message; PLSA; Multilingual; Useless words

I. INTRODUCTION

With the development of information technology, instant messaging (IM) has become the main manner of daily communication. For instance, as the largest IM platform, there were over one billion registered QQ users till today, and at least 0.7 billion of them are online simultaneously [1]. However, as a consequence of the dispersion of topics and meaningless chatting, online IM groups are filled with useless messages which usually make users miss the important topics and information. As a typical representative of online IM groups, QQ group leaves users at a loss facing tens or hundreds of messages unread when they just leave for a coffee. Topic detection may help IM users capture what the IM group is talking about without reading all the messages [2]–[6]. Unfortunately, it is not an easy task, in particular in the case where multiple languages are used and new terms keep emerging. In this scenario, the traditional topic detection methods designed for single-language documents with limited terms are no longer applicable. In this work, based on the Probabilistic Latent Semantic Analysis (PLSA), we design a new method, aiming at finding useful topic information from hundreds of multilingual messages efficiently.

A. Related Work

Topic detection in instant messages differs from the topic detection in documents due to the following three facts [7]:

1) the instant messages may contain large amount of useless terms, which would keep emerging during the online chatting; 2) compared with the documents, the instant messages are usually much shorter, containing only a few words per message; and 3) it is also possible that multiple languages would be used alternately. Therefore, compared with topic detection in documents [8], topic detection in instant messages is a challenging research task. Relatively only a few methods have been proposed [2]–[5], [9].

Kolenda et al. [2] utilized Independent Component Analysis (ICA) combined with Latent Semantic Analysis (LSA) for chat room topic detection. They first remove stop words from chat messages. Then they partition the messages into “sessions” and apply LSA over the “sessions”. Finally ICA is carried out to analyze each session. However, the result is greatly influenced by the partition of messages. Moreover, LSA lacks a solid statistic foundation and has a high time complexity. Similarly, Bingham et al. [3] proposed another method for topic detection in a chat room scenario. The difference is that ICA is replaced by Complexity Pursuit (CP) which partitions the messages by time. However, this method has the same shortcomings as the method of Kolenda et al., where in online IM group, conversation can be intermittent which decreases the accuracy of the partition work.

Aurora et al. [4] proposed a hierarchical-clustering-based method to cluster news in a news stream and summarize the topics of each hierarchy. Then the user can select processed topics. They succeed to reduce the time complexity of the algorithm and the method also works for multilingual data. However, when it comes to online IM groups, the topics are not in a hierarchical structure and it’s unreasonable to apply hierarchical clustering over the messages. In [5], Paige H. Adams and Craig H. Martell proposed a TF-IDF based method for topic detection and extraction in chat. They compute the similarity of posts in a message stream by simple cosine similarity, and utilize a combination of four techniques: TF-IDF, hypernym augmentation, nickname augmentation, and time-distance penalization. However, this method matches posts on a text level without a deeper consideration of semantic information.

B. Our Work

To address the above issues, in this work, we propose a new method based on PLSA, which is capable of efficiently...
capturing meaningful topic information from noisy, short instant messages. The basic step is to treat each message in an online group discussion as a data item in message stream, and then apply PLSA on the collected text messages. Instead of applying Machine Translation (MT) over the messages to translate multilingual material into single language, we use the corresponding word-division method to cut a message into words of different languages, for instance, performing MMSEG [10] for Chinese and for English, taking each word separately. The stop-word list is a list of those meaningless words or words without stand-alone semantics in different languages. For example, “am, is, are, was, were, uh-huh, a...” in English. After removing meaningless words, each message is transformed into a bag of meaningful words, which is used to form a message-word co-occurrence matrix.

By further performing PLSA on the resulting message-word co-occurrence matrix, we can get the probability of each text message and the latent topics. Probabilistic Latent Semantic Analysis (PLSA), also called Probabilistic Latent Semantic Indexing (PLSI), is a technique widely used for modeling topics [11]. In our work, we utilize it to model the message-word co-occurrence matrix information under the probabilistic framework so as to discover the underlying topic structure of the message streams. We choose PLSA because the model suits the IM situation best with its convincing assumption, the latent variable z which can be naturally and reasonably regarded as the parameter reflecting the topic importance. Unlike the standard Latent Semantic Analysis (LSA) [12], which is only based on linear algebra, i.e. singular value decomposition (SVD), PLSA possesses a sound probabilistic interpretation, which makes it superior to LSA. As a result, our method has a solid statistical foundation to find the probability of any message by enumerating every possible topic.

Since each message is treated as a collection of words without considering the order of them, there is no need to consider whether the message is written in a standard language (such as English) as long as it can be divided into words. When there’re few messages about a topic, it contributes litter to the entire discussion and then in the final result, these messages may not appear because this topic can be considered as noise. The result of our approach to some degree depends on the initialization of some parameters. In our approach, the parameters are initialized randomly and we compared the performance of different initialization of parameters in our experiments.

The rest of the paper is organized as follows. In section II, the proposed method is described in detail. Section III reports the experimental results. We conclude our paper in section IV.

II. TOPIC DISCOVERY IN MULTILINGUAL MESSAGE STREAM

The proposed approach consists of two main steps. The fist step is to construct a multilingual message-word co-occurrence matrix from the instant message stream. The second step is to use PLSA to find the importance of each message so as to find the most influential messages as output to users.

A. Multilingual Message Stream Preprocessing

Given an instant message stream consisting of M messages \( S = \{ s_i : i = 1, \ldots, M \} \), the first step of our method is to construct a multilingual message-word co-occurrence matrix. To this end, we need to firstly segment each message into a bag of words by using word-division methods. That is, for some language terms such as Chinese characters, some specific word-division method should be applied in order to get meaningful words; and for the other language terms such as English characters, direct word-division based on space is suitable.

Then a method is designed to remove the stop-words. Stop-words are those meaningless words which makes little contribution to the meaning of the message. Some supervised learning methods can be applied to obtain a chatting-stop-word list. For each message, after we segment it into a bag of words, we examine whether the words contained are in the stop-word list and remove the words that are in the list.

After this preprocess, each message \( s_i \) is transformed into a bag of meaningful words. Assume that there are in total \( N \) unique meaningful words in the \( M \) messages, denoted as \( W = \{ w_j : j = 1, \ldots, N \} \), a multilingual message-word co-occurrence matrix of size \( M \times N \) can be constructed, with the \( M \) rows representing \( M \) messages and the \( N \) columns representing \( N \) unique words. An observed pair \( n(s_i, w_j) \) in the \( i-j \) entry represents the occurrence of a word \( w_j \) in a particular language in the multilingual message \( s_i \), that is, the number of times \( w_j \) appears in message \( s_i \). Notice that, in the preprocessing step above, the terms in different languages are considered as the same type of “object”, say “words”. Therefore, the problem associated with multilingual message can be well addressed.

B. Topic Discovery Based on PLSA

Assume that the IM group discussion is about \( k \) topics \( Z = \{ z_1, \ldots, z_k \} \), where the number of topics \( k \) is known as a priori. The messages \( S \), latent topics \( Z \) and the words \( W \) are linked via a graphical model as shown in Fig. 1. To begin with, some probability notations are firstly introduced below.

- \( P(s) \) : the probability of selecting a message \( s \).
• \( P(z|s) \): the probability of selecting a topic \( z \) conditioned on the given message \( s \).

• \( P(w|z) \): the probability of selecting a word \( w \) conditioned on the given topic \( z \).

The objective of PLSA is to maximize the following log-likelihood function

\[
\ell = \sum_{s \in S} \sum_{w \in W} n(s, w) \log P(s, w)
\]

(1)

where

\[
P(s, w) = P(s) \sum_{z \in Z} P(w|z)P(z|s) = \sum_{z \in Z} P(z)P(s|z)P(w|z)
\]

(2)

based on Bayes’ rule and conditional independence assumption (i.e. words and documents are conditionally independent given the topic).

In the above objective function, the parameters are \( P(w|z) \) and \( P(z|s) \), which can be estimated by applying the EM algorithm consisting of two steps as follows:

1) E step:

\[
P(z|s, w) = \frac{P(z)P(s|z)P(w|z)}{\sum_{z'} P(z')P(s|z')P(w|z')}
\]

(3)

2) M step:

\[
P(w|z) = \frac{\sum_{s, w} n(s, w)P(z|s, w)}{\sum_{s, w'} n(s, w')P(z|s, w')}
\]

(4)

\[
P(s|z) = \frac{\sum_{w} n(s, w)P(z|s, w)}{\sum_{s', w} n(s', w)P(z|s', w)}
\]

(5)

\[
P(z) = \frac{\sum_{s, w} n(s, w)P(z|s, w)}{R}
\]

(6)

where \( R = \sum_{s, w} n(s, w) \)

Thus we obtain the message-topic matrix. Then we obtain the max estimated probability of each message as follows

\[
P(s) = \sum_{z \in Z} P(z)P(s|z).
\]

(7)

The output messages are top \( k' \) messages according to \( P(s) \), where \( k' \) is determined by user, e.g. with priori knowledge. The iteration ends when the top \( k' \) messages don’t change or the likelihood \( \ell \) changes less some threshold between two consecutive iterations. Algorithm 1 summarizes the proposed method.

III. EXPERIMENTS

In this section, extensive experiments were conducted to analyze the performance of our method from different perspectives as follows:

• The influence of random initialization.

• The analysis of parameters \( k \) and \( k' \).

• The length of stop-word list.

Algorithms

Algorithm 1 Topic detection in instant messages

1: Input: \( k, k', err \).
2: Load stop-word list
3: Load message stream \( S \)
4: for \( \forall s \in S \) do
5: Divide \( s \) into words and remove stop-words
6: end for
7: for \( \forall s \in S \) do
8: for \( \forall z \in Z \) do
9: for \( \forall w \in W \) do
10: Initialize \( n(s, w) \)
11: Random initialize \( 0 < P(s|z) < 1, 0 < P(w|z) < 1 \), and \( 0 < P(z) < 1 \)
12: Initialize \( P(z|s, w) = 0 \)
13: end for
14: end for
15: end for
16: repeat
17: E step: update \( P(z|s, w) \)
18: M step: update \( P(z), P(s|z), P(w|z) \)
19: Re-estimate \( \ell \)
20: until Top \( k' \) messages don’t change or \( \ell \) changes less than \( err \)
21: for \( \forall s \in S \) do
22: Initialize \( P(s) = 0 \)
23: for \( \forall z \in Z \) do
24: \( P(s) = P(s) + P(s|z) \cdot P(z) \)
25: end for
26: end for
27: Sort \( P(s) \)
28: Output: TOP \( k' \) messages by \( P(s) \)

• The proportion of different languages (to analyze the performance of our approach in different multilingual environments).

• The performance over short-message streams.

The experiment data were collected straight from chatting records of QQ groups which are stored as text files. We conducted the experiments by implementing our method in Python running on the ordinary dual core PC. And we developed a method to evaluate the results of our experiments, then analyzed the performance in different situations.

A. Experiment Evaluation

For experimental evaluation, the message stream \( S = \{ s_i : i = 1, \ldots, M \} \) consisting of \( M \) messages is first manually segmented into \( k \) topics, and each topic is scored as \( V_t, \forall t = 1, \ldots, k \) according to its importance among \( k \) topics. For each message \( s_m \in S \), we score it manually by its contribution to the associated topic \( t_m \) among all the messages of the topic, denoted as \( V_{m} \). A high score means this message contributes much to the topic or it’s closely related to important messages. We define the Semantic Value (SV) of a message \( s_m \) as the product of the score of the associated topic \( V_{t_m} \) times the score of its score \( V_{m} \) in the topic,

\[
\lambda_m = V_{t_m} \times V_{m},
\]

(8)
and we evaluate the final topic detection result by accuracy defined as follows

\[ \eta = \frac{n_h}{n_0} \times \frac{\sum_{m \in \delta} \lambda_m}{\sum_{s_i \in \Theta} \lambda_i} \] (9)

where \( \delta \) is the set consisting of the top \( k' \) messages obtained by our approach, \( \lambda_m \) is the \( SV \) of the message \( s_m \); and \( \Theta \) is the set consisting of the ground-truth top \( k' \) messages, \( \lambda_i \) is the \( SV \) of the message \( s_i \), \( n_h \) is the number of topics hit by our method and \( n_0 \) is the total number of topics that determined by human.

Obviously, the \( \eta \) of a test case can’t reach very high because in an important topic, there will be several messages which are scored very high. However, our algorithm will mistakenly find messages in the less important topics, while miss some important messages in the more important topics. As a result, the final \( \eta \) will be around 50%.

B. Experiment Result

This subsection reports the performances of our method from different perspectives. The output of our method is a set of messages which are the most important in each topic, as shown in Fig. 2.

1) Different Random Initialization: We first analyze the performance of our algorithm with different random initializations of \( P(s|z) \), \( p(w|z) \) and \( p(z) \). In this part of experiment, a multilingual message stream consisting of 300 messages are used, with the most suitable parameters \( k = 11, k' = 10 \).

Figure 3 reports the results generated by our method with different random initializations in 10 runs. In this figure, both the iteration number and accuracy are reported. The detailed results of 10 runs are reported in the two upper tables of this figure, and the average performances are reported in the bottom table. From the figure, we can see that, although the number of iterations would strongly depends on the initialization, yet the accuracy performances are relatively stable over different initializations.

2) Parameter Analysis: To analyze the performance of our method with respect to different numbers \( k \) and top message numbers \( k' \), we changed the value of \( k \) (the ground-truth value is 11) and \( k' \) (the ground-truth value is 10) respectively and performed our algorithm over a multilingual message stream with 300 messages. In this part of experiments, the probabilities \( p(s|z) \) and \( p(w|z) \) were initialized evenly over all runs. The results of each test case are shown in Fig. 4 and Fig. 5, which report the results with varying \( k \) and \( k' \) respectively.

As Fig. 4 shows, our algorithm guarantees a relatively stable performance. However, when \( k \) is very small, say, 2, the number of required iterations is also very small, which is very obvious. The topic detection accuracy \( \eta \) also depends on the number of topics. In particular, when the number of topics \( k \) is quite large, say 20, the accuracy decreases to 32.72%.

Figure 5 shows the results with varying \( k' \) but fixed \( k \). Similarly, the performance depends on the number of top messages \( k' \). That is, as \( k' \) grows, the total iteration number increases dramatically, but the accuracy \( \eta \) decreases relatively slowly.

3) Stop-word List Length: We compared the performance of our algorithm when using stop-word lists of different lengths
in the step of useless word removal. The experiment is based on a stream with 200 messages and the result is shown in Fig. 6.

It shows that by increasing the length of stop-word list, the accuracy grows and to a certain extent decreases the scale of data.

4) Language Proportion: We collected 12 conversations where half of them are mainly English and the other half are mainly Chinese. Then we mixed them up in different proportions and performed our algorithm over the mixed message stream with different proportion of languages. The result is shown in Fig. 7.

As the result shows, different language proportion of corpus does not greatly influence the result. That is because we treat messages in any language the same as a collection of words. Different language words and different grammars does not make a difference.

5) Short Messages: We collected conversations with both short and long messages. Then we divided them into two parts, a short-message part whose average length is 5.41, a long-message part whose average length is 10.12, and performed our algorithm over them. The result is shown in Fig. 8.

The performance of our algorithm declines when the average length of messages decreases. However, it can still generate a good result.

IV. CONCLUSION

In this paper, a new topic detection method has been proposed for efficiently extracting impotent topic information from instant messages. The proposed method is based on PLSA. Extensive experiments have been conducted on the real-word chatting data to demonstrate the effectiveness of our method, which show that our method is applicable for the case where 1) useless terms keep emerging; 2) the instant messages are very short; and 3) multiple languages are used.

In our future work, we want to implement our method as a software with more convenient user experience. For instance, by collaborating with Tencent, this method can be developed as a plug-in component in Tencent QQ. Additionally, the
offline-online strategy will also be utilized to provide real-time user experience. That is, the topic detection algorithm will be run as an offline plug-in component in instant messaging platforms, which can automatically process the chatting history and store the topic detection results in disk; Once the users want to explore one specify online group, just browse the topic detection results stored in advance and enjoy a real-time experience.

ACKNOWLEDGEMENT

This work is sponsored by NSFC (61173084), National Science & Technology Pillar Program (No. 2012BAK16B06), CCF-Tencent Open Research Fund, and Research Training Program of SMIE of Sun Yat-sen University.

REFERENCES