Instant Message Clustering Based on Extended Vector Space Model*

Le Wang, Yan Jia, and Weihong Han

Computer School, National University of Defense Technology, Changsha, China
wanglelemail@163.com,
jiayanjy@vip.sina.com,
hanweihong@gmail.com

Abstract. Instant intercommunion techniques such as Instant Messaging (IM) are widely popularized. Aiming at such kind of large scale mass-communication media, clustering on its text content is a practical method to analyze the characteristic of text content in instant messages, and find or track the social hot topics. However, key words in one instant message usually are few, even latent; moreover, single message can not describe the conversational context. This is very different from general document and makes common clustering algorithms unsuitable. A novel method called *WR-KMeans* is proposed, which synthesizes related instant messages as a conversation and enriches conversation's vector by words which are not included in this conversation but are closely related with existing words in this conversation. *WR-KMeans* performs clustering like k-means on this extended vector space of conversations. Experiments on the public datasets show that *WR-KMeans* outperforms the traditional *k*-means and bisecting *k*-means algorithms.

Keywords: instant messages clustering, k-means, Vector Space Model.

1 Introduction

With the rapid development of internet and communication technology, Instant Messaging (IM, e.g. E-mails, SMS, chats through MSN or ICQ, etc.) on internet or mobile network is widely popularized[1], e.g. more than 160 million short text messages were sent over the National Week holiday in Beijing[2]. Considering economic interest or public security, corporations and governments, which provide this service, have stored instant messages in text database for further analytical and mining applications [1]. Instant message clustering is very useful for analyzing its content characteristic or establishing other mining application.

The most common text processing approach is to represent the documents with vectors. This is so-called vector space model, in which a vector corresponds to one

* This project is sponsored by national 863 high technology development foundation (No. 2006AA01Z451, No.2006AA10Z237).

[1] http://www.instantmessagingplanet.com/

[2] http://english.cri.cn//3100/2006/10/10/63@148745.htm

L. Kang, Y. Liu, and S. Zeng (Eds.): ISICA 2007, LNCS 4683, pp. 435–443, 2007.

document and the dimensions correspond to words in this document. Once the high-dimensional vectors are derived, the major challenge left for document clustering is how to deal with these high dimensional data. However, instant message is extremely shorter than the common document. There are usually only several key words in one instant message, and key words about the message topic are even latent sometimes. The sparsity of key words makes word-frequency based methods inappropriate to measure the similarity among instant messages. Table 1 is an example to demonstrate above problem, where instant messages are all about sports and have considerable similarity with each other. Although IM-1 is similar to IM-2 and IM-3 with 0.71 and 0.58 degree respectively, no similarity exists between IM-2 and IM-3, which conflicts with the reality. So the bag-of-word model and term frequency-based measure are not applicable in instant messages mining.

Table 1. Example to illustrate bag-of-word model vectors and similarity between instant messages according to vector inner product

	ball	basketball	football	foot	IM1	IM2	IM3
IM-1	0	1	1	0	-	0.71	0.58
IM-2	0	2	0	0		-	0
IM-3	1	0	2	1			-

This paper proposes two methods to enhance the description of instant messages to response the problem of sparse key words when clustering on instant messages.

Firstly, we notice that instant message is a kind of semi-structured data, which has source and destination addresses with time stamp. Instant messages sent back and forth among specific persons during some specific time intervals form a conversation, which groups these instant messages into a specific topic. So we combine these messages as one conversation. It is obvious that conversation has more key works and more integral context information than simply single message. Then clustering is performed toward conversations instead of messages.

Secondly, we enhance the content description of a conversation with words, which are not in the conversation but are closely related with existing words in this conversation. Fox example, words ‘ball’ and ‘football’ are added to IM-2, which are not appear in IM-2 but have obvious correlation with the word, ‘basketball’, in IM-2.

In this paper, we propose an instant message clustering method called WR-KMeans, which can automatically scan instant message corpora, construct conversations and enhance traditional TF-IDF model by adding relevant words in conversations. WR-KMeans performs clustering on this evolved model of conversations like k-means [2].

WR-Kmeans method is evaluated and compared with two other well-known text clustering methods which is based on traditional TF-IDF model. HowNet knowledge base is used to quantify the relation strengths between words in conversations during the experiments. Experimental evidence shows that WR-KMeans is significantly outperformed. Furthermore, HowNet is Chinese-English bilingual linguistics, so WR-KMeans and its components can be smoothly transformed to process Chinese [3].

The rest of the paper is organized as follows. We present related works in Section 2 and present WR-KMeans method in Section 3. The experimental results are reported in Section 4. Finally, we conclude the paper and discuss future works in section 5.

2 Related Works

Various methods can be used to confirm the boundary of conversation for different type of instant messages. Conversations can be easily captured by the posting threads in Usenet due to its inherent threaded nature [4]. Methods based on certain relevant patterns are used by Faisal M. Khan [5] to identify chat thread starts in chat-room medium flows. These patterns are made up of several sentences, such as “hi, hey” or “how are you”, which are developed by human experts through observing chat conversations. It is an effective method for medium with very strong interaction like chat-room. Marti A. Hearst utilizes TextTiling algorithm to locate topic boundaries within expository text [6]. This algorithm is designed to separate expository text into paragraphs, and uses lexical analyses based on TF-IDF model to determine topic starting point. Methods mentioned above are mainly based on the content of corpora.

The relationships among words have been widely studied in fields of nature language processing, text mining and information retrieval, etc. One method is Latent Semantic Indexing (LSI) [7], which automatically discovers latent relationships among corpora through Singular Vector Decomposition. However, the method is time-consuming when applied to a large corpus. Kenneth Ward Church proposed ‘association ratio’ based on the notion of mutual information to estimate word association norms by their co-occurrence probability [8]. It is not appropriate to refer to words co-occurrence for instant messages because of the key words sparsity. Satoru Ikehara proposed a vector space model based on semantic attributes of words, which uses the Semantic Attribute System [9]. This method aims to reduce the vector dimension using upper-lower relations between semantic attributes of words and achieves good efficiency in processing Japanese.

3 WR-KMeans Method

3.1 Synthesizing Conversation

Message Database (MDB) is a message set, which store messages in a form which facilitates accessing a group of messages. T_1 and T_2 are used to denote the starting and end time of a specific period, respectively. s_i and d_i stand for source and destination addresses. c_i is the text content of instant message. Then the concept of conversation can be formalized as:

Definition 1 (Conversation). $M \in MDB$, $\|M\| = n$, If $\forall m_i =< t_i, d_i, s_i, c_i >$ and $\forall m_j =< t_j, d_j, s_j, c_j >$, $m_i \in M$ and $m_j \in M$, $0 < i, j \leq n$, if m_i and m_j satisfy $T_1 < t_i < T_2$, $T_1 < t_j < T_2$, $s_i = d_j$ or $d_i = s_j$, then the string, $c_1 \mid c_2 \mid \ldots \mid c_n$, synthesizes a conversation between two persons at the interval of T_1 and T_2. ■

Further observation into the instant message data set[3] from the reality mining project of MIT allows us to find that the frequency of IMs transmission before and after one

[3] http://reality.media.mit.edu/dataset.php

conversation is usually lower than that during the conversation, which is shown in Figure 1. So the boundaries of conversations can be determined in some sense by the concaves of frequency. This can be explained as following. From the point of temporal view, peoples tend to densely communicate with each other about the same topic. In other words, instant messages which are produced by one pair of persons and are related to the same topic could be aggregated together approximatively according to their generation time, i.e. the communicating requency.

We define the following rules for synchronizing instant messages into conversations on the basis of above analysis. $V_{i,i+1}$ is used to denotes the time interval between two adjoint instant messages, m_i and m_{i+1}. We assume that if $V_{i,i+1} < \alpha$ and $V_{i,i+1} < V_{i+1,i+2}$, then m_{i+1} and m_i belongs to the same conversation; otherwise, m_{i+1} is the starting of next conversation. Where α is a statistic constant which describes the biggest interval between two adjoint instant messages that belong to the same conversation. *WR-KMeans* orderly compares the intervals of adjoint IMs between two specific persons and synthesizes conversations for each pair of all persons.

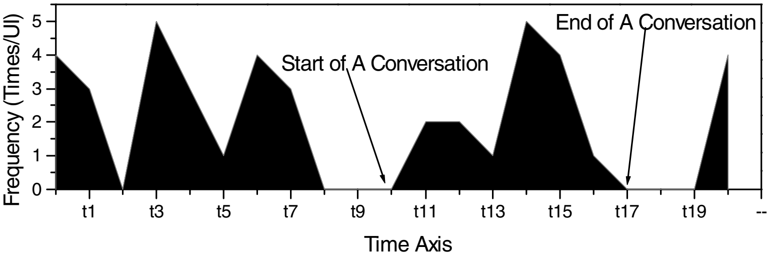

Fig. 1. Frequency Change of IM transmission between two persons in a specific time interval

3.2 Enhancing the Representation of Conversation Through Relevant Words

Assume that there are *m* conversations, which consist of *n* different words totally. We calculate the relevant strength of each pair of words according to *HowNet*. Given a conversation C_l, the word, t_j, which is not in conversation C_l, is used to enhance the vector representation of C_l if the relevant strength ($\delta_{i,j}$) between t_j and t_i, which is originally in C_l, is beyond one threshold of relevant strength.

In addition, the weights of the words in corpora are not equal to each other. The important word in conversations is usually the one which defines few conversations. That is the more irregular word which is the more important for distinguish the conversations. This relies on the information entropy of the word, which is defined as $E_i = -p_i \cdot \log_2 p_i, (0 < i \le n)$ and $p_i = \lambda_i / m$. λ_i is the sum of conversations that include term t_i. Then the weight of word t_i in C_l is defined as formula (1) where $num(C_l)$ is the total num of words in conversation C_l.

$$\beta_i = \frac{E_i}{\sum_{k=1}^{num(C_l)} E_k}, (0 < i \leq num(C_l)) \tag{1}$$

Then the value in vector delegating word t_j, which is added to enhance the vector representation of the conversation, can be determined according to formula (2) where t_k is the value of k-th word in the vector of C_l according to TF-IDF model.

$$t_j = \sum_{k=1}^{k=num(c_l)} \beta_k \cdot t_k \cdot \delta_{k,j} \tag{2}$$

For example, in table 2, a word-by-conversation matrix is constructed from 3 conversations (C1, C2, and C3) and 7 words. Only relevant strengths that are beyond 0.4 are considered and set $\delta_{i,j}$ equal to $\delta_{j,i}$. For T1 in C1, the value in TF-IDF is $2{*}\log_{10}(3/1)=0.9542$. The word, T4, which is not in C1, has a relevant strength beyond 0.4 with T1. So T4 should be added into the vector of C1. The value of T4 in vector is $0.9542{*}0.5283{*}0.62=0.3126$, where $-1/3{*}\log_2(1/3)=0.5283$ is the weight of T1.

WR-KMeans is developed as an instant messages clustering method, which is a variant on standard k-means algorithm. This algorithm preprocesses the instant messages, synthesizes conversations and extendes the vectors of conversations before performing clustering on enhanced TF-IDF model. *WR-KMeans* measures the similarity between conversations according to a cosine measure.

The sum of terms is in a finite bound, preprocesses relate mainly with the volume of instant message set. So *WR-KMeans* is an extensible method.

Table 2. Example of extending word-by-conversation matrix and word relevant strengths obtained through querying of *HowNet*

	Original word										TF-IDF model		
	Frequency			word relevant strengths							enhanced by WR		
	C1	C2	C3	T1	T2	T3	T4	T5	T6	T7	C1	C2	C3
T1	2	0	0	N/A	–	–	0.62	–	–	–	0.9542	0.1277	0.0852
T2	1	1	0		N/A	–	–	0.43	–	–	0.1761	0.1761	0.4336
T3	0	1	1			N/A	–	–	–	0.42	0.2117	0.1761	0.1761
T4	0	3	2				N/A	–	–	–	0.3126	0.5283	0.3522
T5	0	0	4					N/A	–	–	0.0295	0.0295	1.9085
T6	0	2	0						N/A	–	0	0.9542	0
T7	2	0	0							N/A	0.9542	0.0288	0.0288

4 Experimental Evaluations

Three different algorithms, *WR-KMeans*, Bisecting k-means and standard k-means, are implemented and compared. All these experiments are performed on two public datasets with manually predefined categorizations.

4.1 Evaluation Criteria

Two cluster validation methods, Silhouette Coefficient (SC) [11] and normalized mutual information (NMI) [12], are used to evaluate the clustering performance because both of them are independent from the number of clusters, *k*.

Suppose that instant messages are synchronized into m conversations, which have l classes. $C_M = \left\{\overline{C_1}, \overline{C_2}, \cdots, \overline{C_k}\right\}$ defines a clustering result.

Silhouette Coefficient (SC)

$S(C_i, \overline{C_j})$ is the similarity of a conversation C_i to a cluster $\overline{C_j}$, which equals to the average similarity between C_i and each conversation in $\overline{C_j}$. Let $f(C_i, C_M) = S(C_i, \overline{C_j})$ be the similarity between the conversation C_i and its cluster $\overline{C_j}(C_i \in \overline{C_j})$, and $g(C_i, C_M) = \max_{\overline{C_j} \in C_M, C_i \notin \overline{C_j}} S(C_i, \overline{C_j})$, the similarity between C_i and the nearest neighboring cluster. The silhouette of C_i is defined as:

$$SC(C_i, C_M) = \frac{f(C_i, C_M) - g(C_i, C_M)}{\max\{f(C_i, C_M), g(C_i, C_M)\}} \tag{3}$$

The silhouette coefficient is defined as formula (4). Its value is usually between 0 and 1. Values beyond 0.5 indicate that clustering results are separable clearly. If they fall below 0.25, it becomes very difficult to find practically significant clusters.

$$SC(C_M) = \frac{\sum_{C_i \in MDB} SC(C_i, C_M)}{|MDB|} \tag{4}$$

Normalized Mutual Information (NMI)

Let m_i, m_j be the numbers of conversations in the i-th class M_i and j-th cluster $\overline{C_j}$ respectively, m_{ij} is the number of conversations of M_i that are assigned to $\overline{C_j}$, m is the total number of conversations in MDB. NMI is then defined as formula (5), which equals to 1 when clustering results perfectly match the external category labels and is close to 0 for a random partitioning. NMI measures the consistent level between the clustering result and the original classification in data set, the later is usually provided by human experts. The bigger value of NMI is the more ideal consistency with the outcome of human beings, which illuminates a perfect clustering method.

$$NMI(C_M) = \frac{\sum_{i=1}^{l} \sum_{j=1}^{k} m_{ij} \cdot \log(\frac{m \cdot m_{ij}}{m_i \cdot m_j})}{\sqrt{(\sum_{i=1}^{l} m_i \cdot \log \frac{m_i}{m})(\sum_{j=1}^{k} m_j \cdot \log \frac{m_j}{m})}} \tag{5}$$

4.2 Experimental Setting

We use two data sets: (1) the Reuters-21578 corpus[4], (2) 20-newsgroups data[5]. These two datasets comprise priori categorizations of documents, and their domains are broad enough to be as realistic as conversations. We preprocess the raw datasets mentioned above using the Bow toolkit[6] and Porter stemming function [13].

HowNet system version 2000, a free edition of this software, is used to quantify the mutuality between terms. One HP unit with 4 Itanium II 1.6G processors and 48 GB memory is used as hardware platform.

The word-by-conversation matrixes of two datasets are preprocessed according to TF-IDF model (for standard *k*-means and Bisecting *k*-means algorithms) and enhanced TF-IDF model (for *WR-KMeans* method) respectively.

4.3 Experimental Results

We use a maximum number of iterations of 20 (to make a fair comparison) for all these three algorithms. Each experiment is running ten times. We set the threshold of relevant strength between two words to 0.4.

Table 3. NMI results on 20-Newsgroup

k	5	15	20	25
Std k-means	.23±.03	.24±.02	.26±.03	.25±.02
Bis k-means	.37±.02	.40±.02	.42±.01	.41±.03
WR-KMeans	**.54±.03**	**.68±.02**	**.79±.01**	**.72±.02**

Table 4. NMI results on Reuters-21578

k	40	60	80	100
Std k-means	.21±.03	.22±.01	.25±.02	.24±.03
Bis k-means	.32±.03	.36±.02	.40±.01	.38±.02
WR-KMeans	**.46±.03**	**.52±.01**	**.64±.02**	**.58±.02**

Table 3 and Table 4 report the the effect of k on NMI results on NG20 and Reuters-21578, respectively. NMI measures the degree of consistency between clustering results and manually predefined categorizations, i.e. the superposition of clusters and classes. *WR-KMeans* has clear better clustering results, which is indicated by Table 3 and Table 4. The reason is that the extended vector space model has more enriched semantic information than traditional TF-IDF model, and the strengthened vector represents the real theme of text content. The vectors, in which only correlated terms are added, are used to compute the similarities. This approach magnificently avoids the warp resulted from the sparsity of key words when measuring the similarities of text and then achieve the better effectiveness than primal representation method of TF-IDF model.

[4] http://www.daviddlewis.com/resources/testcollections/reuters21578/

[5] http://kdd.ics.uci.edu/databases/20newsgroups/20newsgroups.html

[6] http://www.cs.cmu.edu/mccallum/bow

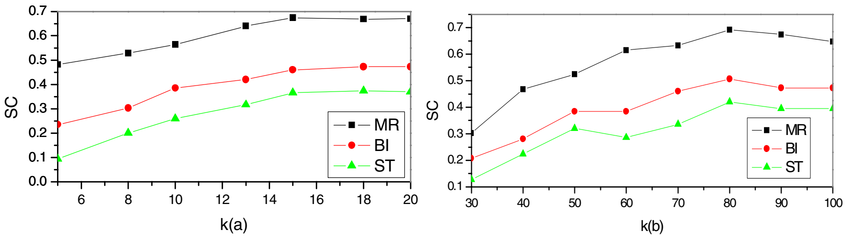

Fig. 1. Comparing the best SC results on NG20(a) and on Reuters-21578(b)

We perform experiments on NG20 and Reuters-21578 to study the effect of k on SC, and the result is shown in Figure 1. The SC studies the divisibility of clusters. At the point of original num of classes in dataset, *WR-KMeans* can get clear partitions of corpora, which can be induced according to SC in figure 1 (NG20 0.67 when k=20, Reuters-21578 0.69 when k=80). This is a reasonable result.

We can draw the conclusion from above results that the extended vector space model, which is combined with term mutual information, has more linguistic knowledge than TF-IDF model. It takes context information to distinguish the category of documents (conversations).

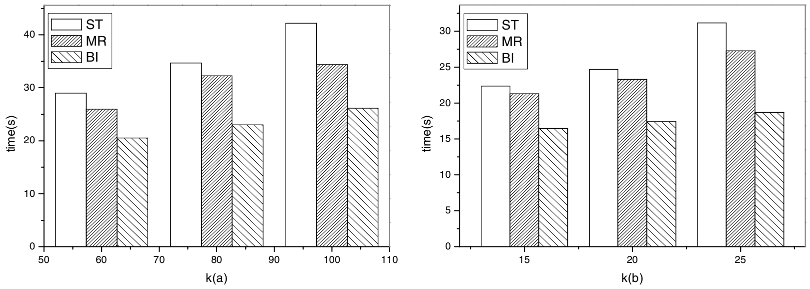

Fig. 2. Comparing the average time on NG20(a) and on Reuters-21578(b)

Figure 2 illuminates the running time of three algorithms on two datasets. We can see that *WR-KMeans* would comparatively needs more time than Bisecting k-means, but a little faster than standard k-means. The reason is that WR-KMeans is optimized only in preprocessing and text representing, not including the clustering process. Although WR-KMeans achieve better effectiveness than other two algorithms, it has not too much advantage in efficiency.

5 Conclusion and Future Works

In this paper, we focus on the instant messages clustering and propose *WR-Kmeans* method to solve the sparsity of key words arising from it. *WR-KMeans* automatically synthesizes instant messages into conversation, which has more key words and more integral context information than simply single message, and extends traditional TF-IDF model of conversations by relevant words by the aid of *HowNet*. Experimental evidence shows that *WR-KMeans* is significantly outperformed against other two method based on traditional TF-IDF model.

We plan to improve the speed of *WR-KMeans* when performing clustering by optimizing its initial partitions. In addition, we want analyze the network of IM by social network analysis in the future works.

References

1. Resig, J., Teredesai, A.: A framework for mining instant messaging services. In: Proceedings of the 2004 SIAM Lake Buena Vista, Florida (2004)
2. MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: proceedings of 5th berkeley SMSP, pp. 281–297 (1967)
3. Guan, Y., et al.: Quantifying Semantic Similarity of Chinese Words from Hownet. In: IEEE Proceedings of ICMLC02, Beijing, vol. 1, pp. 234–239. IEEE Computer Society Press, Los Alamitos (2002)
4. Sack, et al.: A Content-Based Usenet Newsgroup Browser. In: Proceedings of the international conference on Intelligent user interfaces, New Orleans, Louisianna, pp. 233–240 (2000)
5. Khan, F.M., Fisher, T.A., Shuler, L., Wu, T., Pottenger, W.M.: Mining chat-room conversations for social and semantic interactions (2002)
6. Hearst, M.A.: TextTiling: A Quantitative Approach to Discourse Segmentation, Technical Report UCB: S2K-93-24 (1993)
7. Deerwester, S., et al.: Indexing by latent semantic analysis. Journal of the American Society of Information Science 41(6), 391–407 (1990)
8. Ding, C.H.Q.: A probabilistic model for dimensionality reduction in information retrieval and filtering. In: Proc. of the 1st SIAM, Raleigh, NC (2000)
9. Ikehara, S., et al.: Vector space model based on semantic attributes of words. In: Proc. of the Pacific Association for Computational Linguistics (PACLING), Kitakyushu, Japan (2001)
10. Daemi, A., et al.: From Ontologies to Trust through Entropy. In: Proceedings of the International Conference on Advances in Intelligent System, Luxembourg (2004)
11. Hotho, A., et al.: Ontology-based Text Document Clustering. KI 16(4), 48–54 (2002)
12. Strehl, A., Ghosh, J.: Cluster ensembles - a knowledge reuse framework for combining partitions. Journal of Machine Learning Research 3, 583–617 (2002)
13. Porter, M.F.: An algorithm for suffix stripping. Program 14(3), 130–137 (1980)