CIST System Report for SIGdial MultiLing 2015

Shuhong Wan, Lei Li, Taiwen Huang, Zhiqiao Gao, Liyuan Mao, Fang Huang
Center for Intelligence Science and Technology (CIST),
School of Computer Science and Technology,
Beijing University of Posts and Telecommunications (BUPT), China
wsh@bupt.edu.cn, leili@bupt.edu.cn, zhidao2010@bupt.edu.cn, qiaogaozhi@bupt.edu.cn,
753543585@qq.com,xprince@bupt.edu.cn

Abstract

This report provides a description of the methods applied in our CIST system participating two tasks in SIGdial MultiLing 2015. For the task of MMS, we adopt hLDA modeling for documents and sentence extraction to generate a summary. For the task of OnForumS, we utilize Word Embedding Model in deep learning and compute sentence similarity for content linking; then implement LDA topic modeling for argument label and sentiment analysis for sentiment label. According to the published evaluation results, we have got the best performance for both argument label and sentiment label.

1 Introduction

MultiLing (http://multiling.iit.demokritos.gr) is a special session in SIGdial 2015, which holds 4 tasks, i.e., MMS (Multilingual Multi-document Summarization), MSS (Multilingual Single-document Summarization), OnForumS (Online Forum Summarization) and CCCS (Call Centre Conversation Summarization). We have participated in two tasks of MMS and OnForumS. For MMS, we utilize the language-independent hLDA (hierarchical Latent Dirichlet Allocation (LDA)) model to mine a hierarchical topic tree for each document set in all ten languages: Arabic, Chinese, Czech, English, French, Greek, Hebrew, Hindi, Romanian and Spanish. For OnForumS, we adopt the Word Embedding Model to dig up content linking of sentence pair with deeper semantic features. Then we mainly use rule-based sentiment analysis to obtain the sentiment label. LDA topic model and K-means are integrated to obtain the argument label. According to the published evaluation results of OnForumS, we have got the best performance for both argument label and sentiment label.

2 System Background

2.1 MMS

There are many researches about summarization [1-7]. Recently, LDA[8] has been widely applied [9-10]. Some improvements have also been made [11-14]. [15] extended LDA to exploit the hierarchical tree structure of topics, hDLA, which is unsupervised method in which topic number can grow with the data set automatically. This can achieve a deeper semantic model similar with human understanding and is especially helpful for summarization. [16] provided a multi-document summarization method based on hLDA with competitive results. However, it has the disadvantage of relying on ideal summaries. To avoid this, the innovation of our work is completely dependent on data and hierarchy to extract candidate summary sentences.

2.2 OnForumS

Online forums have been increasing greatly in recent years with a large number of users and huge amount of information. This task has really proposed a very new and interesting topic for research. After analyzing the sample input and output data file released by the organizers of OnForumS, we confirmed that OnForumS task involves two parts: The first part is to find all the linking pairs of sentences. In every pair, one sentence belongs to the original article by an author, the other belongs to its comment by a later commentator. The second part is to tag two kinds of labels to the linking pair which we've found in the first part. Labels involve argument label and sentiment label. For argument label, we think that it focuses on whether or not a commentator holds the same argument with the author, hence we use LDA to model the document set (every sentence is a processing unit) so as to mine the latent semantics of sentences. Next, we utilize the vector of topic weight generated by LDA and K-means algorithm to cluster the documents into

two categories corresponding to two argument labels. For sentiment label, we think that it cares about the sentiment of sentences, so we do sentiment analysis based on sentiment dictionaries. We use three English dictionaries as seed, and adopt a machine learning way to expand them. At last, we make some rules to generate our final sentiment label according to some experiments.

2.2.1 Content Linking

The research for the comments have been carried out since they appeared, but it was mainly aimed for the opinion mining of product pages in the past, focusing on the study of emotional tendency[17]. Consumers, manufacturers and retailers can get the feedback they need, using the results of data mining and analysis of comments for commodities[18].

We look on content linking as finding two closely related sentences based on similarity computing. The result of content linking we've got is not optimistic through traditional feature-based similarity calculation. So we attempt to bring in deep learning technology applied to big data --- Word Embedding method to strengthen the traditional methods based on grammars and shallow semantics. The classic work is by Bengio et al. [19]. They got word vector while training language model. [20] proposed a language model of Log-Bilinear and a hierarchical idea to improve Bengio's method. [21] proposed two major innovations: one is using the global information to assist the existing local information, the other is using multiple word vectors to represent polysemous words.

This representation learning is also applied to a variety of natural language processing tasks with excellent results, such as Chinese word segmentation [22], semantic modeling and sentiment analysis [23], named entity recognition [24].

2.2.2 Labels

For argument label, we consider it as a classification problem. We can find some exploration from OpinionFinder System (MPQA, http://mpqa.cs.pitt.edu/). It is based on the corpus marked by experts. Considering this condition, we use unsupervised ways to find the topic of each sentence in the sentence pair, and if the two sentences in a pair are about the same topic, we endow argument label with positive, else, we emend it with negative.

For sentiment label, we regard it as a sentiment analysis problem. There are many researches about text sentiment analysis, which is always a research hotspot. The existing methods can be divided into four levels. The first level is for lexical items; the second level is for sentences; the third level is for texts and chapters. The last level is for massive text data. In our system, we focus on the first two levels. [25, 26, 27] put forward some ways to calculate the sentiment polarity of words. [28, 29, 30] put forward models to analyze text sentiment. In addition, [31, 32, 33, 34, 35, 36, 37, 38] also get some breakthrough in text sentiment analysis.

[39] puts forward Double LDA(DLDA) to solve the problem in analysis of sentiment polarity of texts. DLDA is based on LDA model which is only considering topic, however, DLDA is also involved in sentiment. DLDA has two processes, the first one is called DLDA-Ⅰ, the second one is called DLDA-Ⅱ. By DLDA-Ⅰ, we can get the sentiment distribution of documents and the topic distribution of documents. If we only use LDA, we can simply get the topic distribution. After DLDA-Ⅱ, the result we can get is every word's sentiment weight in all documents.

DLDA has many advantages and we decide to use it.

1) It's unsupervised method. For our task, training corpus is not adequate, thus DLDA is fitful for our task.

2) It's language independent model. The model is similar to LDA. For our task, we need to handle two languages (English and Italian).

3) Many other similar models cannot get the word sentiment weight, they only give the distribution of documents over sentiment.

4) We need to expand our seed sentiment dictionary. We adopt DLDA to expand our seed dictionary from MPQA. We mainly use subjectivity Lexicon and the lexicons in OpinionFinder System.

3 MMS System Design

The system framework for MMS is shown in Figure 1. In particular, we only treat Chinese with word segmentation. The kernel module is constructing an hLDA model.

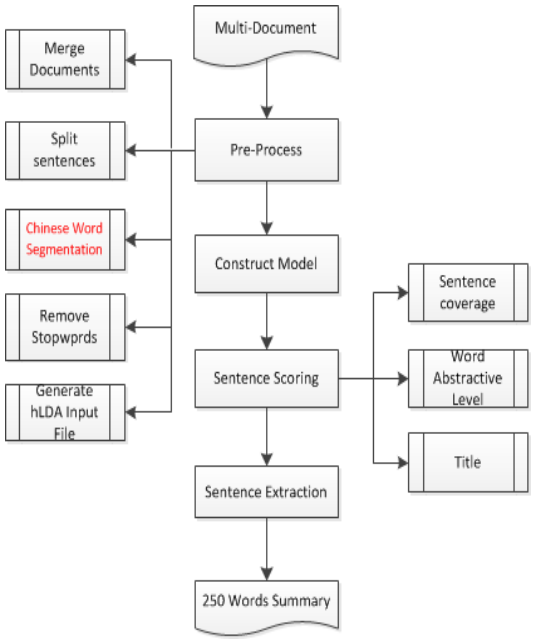

Fig.1. MMS system framework

3.1 Pre-processing

1. Merging documents

We merge the multi-documents into one big document.

2. Splitting Sentence

Sentence is the processing unit. There are two lines of title and date ending with no punctuation marks. We add a full stop to avoid them being connected with the first following sentence. We split sentences according to the ending punctuation marks like full stop, question mark and apostrophe.

3. Removing Stop Words

Since Chinese hanzi is not separated by spaces as other languages, we do word segmentation firstly. Then we construct stop lists for all languages. For English and Chinese, the stop list contains punctuation marks and some functional words, while for other languages, it contains punctuation marks, which could unify the whole process easily.

4. Generating Input file for hLDA

We build a dictionary for the remaining words, which are sorted in descending order according to their frequencies. This is a mapping from a word to a number varying from 1 to dictionary size. Finally we generate an input file for hLDA, in which each line represents a sentence:
[number of words in the sentence] [wordNumber A]:[localfrequencyA] [wordNumberB]:[local frequencyB]...
Figure 2 shows an example.

```
2  11 121:1 319:1 1448:1 904:1 27:1 282:1 804:1 3
.  6:1 1474:1 4:1 106:1
3  20 3:1 313:1 922:1 282:1 816:1 36:1 1538:1 4:1
.  769:1 753:1 1031:1 208:1 569:1 244:1 1163:1 66
.  7:2 1156:1 5:2 72:1 1479:1
4  17 822:1 19:1 34:1 484:1 488:1 143:1 466:1 282
.  :1 300:1 4:1 55:1 1052:1 40:1 310:1 1455:1 38:
.  1 53:1
5  12 139:1 59:1 1401:1 3:1 319:1 50:1 19:1 8:1 6
.  65:1 345:1 677:1 100:1
```

Fig.2. hLDA input file

3.2 hLDA Topic Modeling

In hLDA, every sentence is allocated to a path from the root to the leaf in the tree. Each node is associated with a latent topic, which is a distribution across words. Sentences sharing the same path should be similar to each other and form a sub-cluster of sentences in the document set. All sentences share the topic distribution associated with the root node.

We set the depth of the tree as 3. Because a sentence of 2 levels seems too simple, and a tree model with depth 4 or bigger is too complex for computing and understanding.

Different parameters lead to different trees. We can evaluate whether a tree is good by human reading. However we don't know other languages except Chinese and English, hence we design a simpler and more intuitive method for the hLDA tree evaluation. If a tree has about 4 to 12 paths and the sentence numbers for all paths appear in balanced order from bigger to smaller, and the sentences in bigger paths could occupy 70-85% in all sentences, then we could possibly infer that this model performs well.

There are several parameters in hLDA, which are ETA, GAM, GEM_MEAN, GEM_SCALE, SCALING_SHAPE and SCALING_SCALE. After several experiments, the parameter settings are as follows in Table 1:

Parameter	settings
ETA	1.2, 0.5, 0.05
GAM	1.0,1.0
GEM_MEAN	0.5
GEM_SCALE	100
SCALING_SHAPE	1
SCALING_SCALE	0.5

Table 1: Parameter settings

We analyze the hLDA output result, and change parameters a little when the result is not good.

3.3 Sentence Extraction

- Sentence Scoring

In the hLDA result, sentences are clustered into sub-topics in a hierarchical tree for a document set. A sub-topic is more important if it contains more sentences. Trivial sub-topics containing only one or two sentences can be neglected. Summary should cover those most important sub-topics with their most representative sentences. We evaluate the sentence importance considering the following 3 features.

1) Sentence Coverage. We consider sentence coverage of each word in one sentence. The sentence weight is calculated as eq.(1).

$$S_{tf} = \frac{\sum_{i=1}^{|s|} \frac{num_s(w_i)}{n}}{|s|} \quad (1)$$

w_i is the i[th] word in the sentence, while $num_s(w_i)$ is the number of sentences that w_i occurs. |s| is the number of words in the sentence, and *n* is the total number of all sentences.

2) Word Abstractive level. In hLDA model, level 0 (the root) is the most abstractive one, level 2 (the leaves) is the most specific one, and level 1 is between them. We evaluate the sentences' abstractive features as eq.(2).

$$S_{ab} = a \times \frac{num(w_0)}{|s|} + b \times \frac{num(w_1)}{|s|} + c \times \frac{num(w_2)}{|s|} \quad (2)$$

num(w_0), num(w_1) and num(w_2) are the number of words of the sentence in level 0,1 and 2. Three parameters *a*, *b* and *c* controls the weight of different levels. We need the summary not only provide abstractive information but also specific information. In our experiment, *a*, *b* and *c* are set to be 1, 0.75 and 0.25 respectively.

3) Title. Each document has a title, which is the news reporter's summary statement on the content, it has bigger possibility of being extracted into the summary.

$$S_t = \begin{cases} 1 & the\ sentence\ is\ a\ title \\ 0 & else \end{cases} \quad (3)$$

According to the above features, we calculate the score of a sentence as eq.(4), where *d*, *e*, *f* are the feature weights:

$$S = \mathrm{d} \times S_{tf} + e \times S_{ab} + \mathrm{f} \times S_{\mathrm{t}} \quad (4)$$

In our system, we set *d*, *e*, *f* to be 2, 1, 0.5 respectively.

- Extraction Strategy

We made 3 different strategies to extract candidate summary sentences.

1) The sentences of each path are in descending sequence by score. Extract the sentence which has the biggest score according to its path.
2) All sentences are in descending sequence by score. Extract the sentence which has the biggest score.
3) The above two methods all extract sentences in the summary directly, but the third method chooses enough candidate sentences according to its path. Then delete some sentences based on score to make the length not exceeding 250.

All the extractions should make sure that the sentences' redundancy rate is below 0.5.

4 OnForumS System Design

Figure 3 shows the overall framework for our OnForumS system.

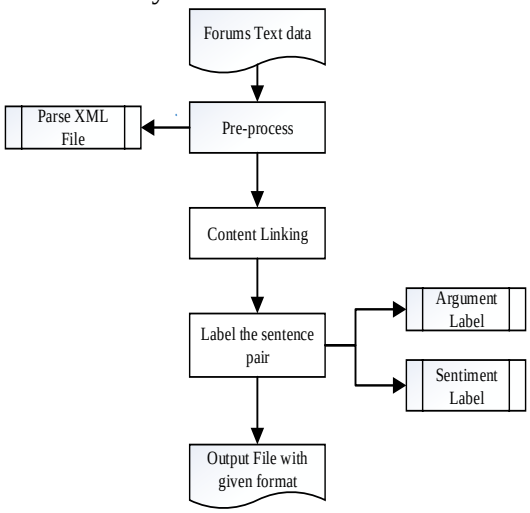

Fig3. OnForumS system overall framework

4.1 Pre-processing

The original corpus of forums has two formats, i.e., txt and xml. Since xml format is easier to handle, we use DOM which is a toolkit in java to parse the xml file to get the all the sentences.

4.2 Content Linking

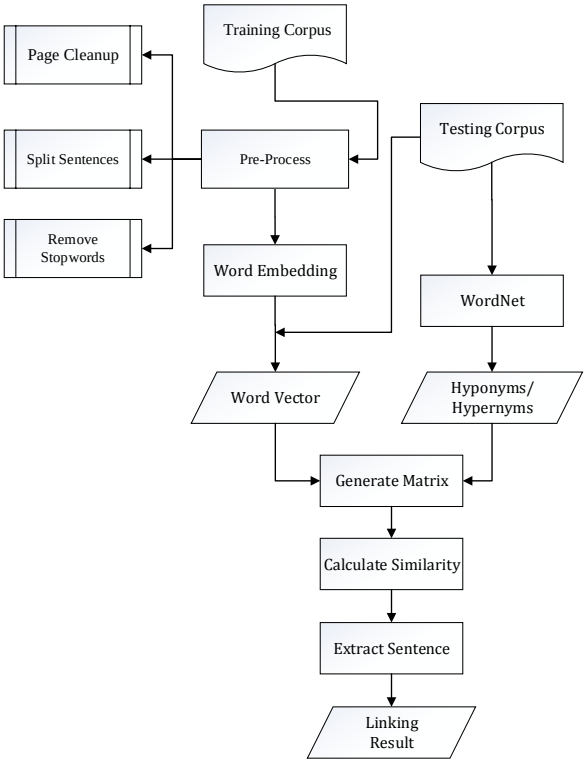

Fig4. Content Linking process

Figure 4 shows the process of content linking based on Word Embedding model and WordNet toolkit.

A. Pre-Processing

Word Embedding Model training needs a large amount of textual data, but both of the sample data and the testing data of OnForumS are too small in size. So we tried to collect the data ourselves for training corpus from Wikipedia (http://en.wikipedia.org/wiki/Wikipedia:Database_download) via a crawler. The size of our final training corpus is about 1G. The next task is page cleaning and re-encoding. Then we split paragraphs into sentences by some punctuations, such as ".", "!", "?", and split sentences into words by spaces.

B. hyponyms/ hypernyms

WordNet is a semantics-oriented dictionary of English, similar to a traditional thesaurus but with a richer structure, which makes it easy to navigate between concepts. For example, given a concept like "car", we can look at the concepts that are more specific—the hyponyms: "Stanley steamer", "hardtop", "loaner" and so on. We can also navigate up the hierarchy by visiting hypernyms, like "car": "motor vehicle".

C. Word Embedding Model

GloVe[40] is a new global log-bilinear regression model for unsupervised learning of word representations and uses the statistics of word occurrences in a corpus whose statistics are captured directly. The calculation equation is as followed:

$$\mathrm{F}(w_i, w_j, \tilde{w}_k) = \frac{P_{ik}}{P_{jk}}$$

Let the matrix of word-word co-occurrence counts be denoted by X, whose entries X_{ij} tabulate the number of times that word j occurs in the context of word i. Let X_i be the number of times that any word appears in the context of word i. Let $P_{ij} = \mathrm{P(j|i)} = X_{ij}/X_i$ be the probability that word j appear in the context of word i. Noting that the ratio P_{ik}/P_{jk} depends on three words i, j, and k. And $w \in \mathbb{R}^d$ are word vectors and $\tilde{w} \in \mathbb{R}^d$ are separate context word vectors.

Through a series of operation and simplification, adding an additional bias $\tilde{b}_k$ for $\tilde{w}_k$ restores the symmetry, finally we get the following equation:

$$w_i^T \tilde{w}_k + b_i + \tilde{b}_k = \log(X_{ik})$$

It proposes a new weighted least squares regression model as following:

$$\mathrm{J} = \sum_{i,j=1}^{V} f(X_{ij})\,(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - logX_{ij})^2$$

Where V is the size of the vocabulary and $f(X_{ij})$ is the weighting function.

For the word vector generated by GloVe, the Euclidean distance (or cosine similarity) between two word vectors provides an effective method for measuring the linguistic or semantic similarity of the corresponding words.

D. Calculate similarity

After the training of word embedding models, a sentence in the testing corpus can be expressed as:

$$W_t = (w_t, w_{t+1}, \ldots, w_{t+k})$$

Where w_t is word vector of corresponding word t. Then the sentence W_i and the sentence W_j can form calculating matrix $M_{i,j}$:

$$\mathrm{M}_{i,j} = W_i W_j^T = \begin{bmatrix} w_t w_v & \cdots & w_t w_{v+l} \\ \vdots & \ddots & \vdots \\ w_{t+k} w_v & \cdots & w_{t+k} w_{v+l} \end{bmatrix}$$

But before the computation of (w_t, w_v), first, stemmed words are generated and examined in consistency. Second, it is essential to check relations between word t and word v by WordNet. When word t and word v exist in the hyponyms/ hypernyms part of each other, they can be seen as the same.

The cosine distance can represent (w_t, w_v) , and the similarity of sentences i and j is as followed:

$$Sim_{i,j} = \frac{\sum_{m=i,n=j} max(\mathrm{M}_{m,n})}{\sqrt{length_i length_j}}$$

Where $max(\mathrm{M}_{m,n})$ is obtained through the following concrete steps. First, find out the maximum of $\mathrm{M}_{i,j}$, then delete the row and column of the maximum. Next, find the maximum of the remaining matrix and remove row and column like the former step. Do the same procedure until the matrix is empty. Finally add up all the maximum values. $length_i$ represents the number of word vector in the sentence, and $\sqrt{length_i length_j}$ is used to reduce the influence of sentence length.

E. Linking result

Based on the above sentence similarities, we can extract those sentences with the highest similarity score to a comment sentence as its linking result.

4.3 Argument Label

Figure 5 shows the process for argument label.

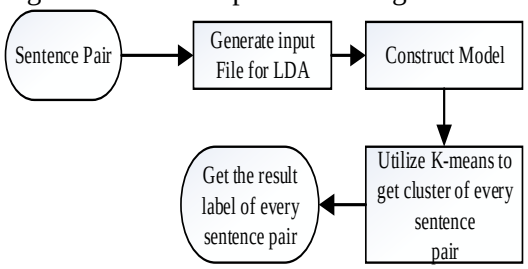

Fig5. Argument label process

4.3.1 Generating Input File For LDA

For every sentence, we change it into its "bag-of-words" model representation, which assumes that the order of words can be neglected. The format of LDA input file is same as that of hLDA input file.

4.3.2 LDA Topic Modeling

Given a collection of sentences in the input file, we wish to discover topic distribution of every sentence through LDA model.

The basic idea of LDA is that documents are represented as random mixture over latent topics, where each topic is characterized by a distribution over words. Through LDA model, we can get the distribution of sentence over topics and the distribution of topic over words. We set the topic number to 15 according to the experiments, that's to say in K-means, our feature is the 15-dimention vector.

4.3.3 K-means and result

K-means is a simple clustering algorithm. Through LDA modeling, every sentence is represented by a vector which is the sentence distribution over topics. This distribution is the input for K-means. We run K-means to cluster all sentences into two categories. After K-means process, we can get the category every sentence belongs to. For every sentence pair, if the two sentences belong to the same category, then we set the label to positive, else, negative.

4.4 Sentiment Label

Figure 6 shows the process for sentiment label.

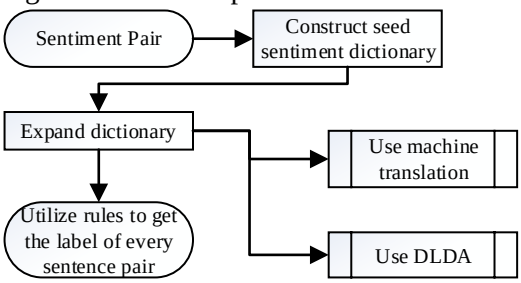

Figure6. Sentiment label process

4.4.1 Seed sentiment dictionary

There are three kinds of seed sentiment dictionaries. One is subjectivity lexicon, the other two are discovered from OpinionFinder system. The two dictionaries are called intensifier and valenceshifters lexicon. Intensifier lexicon involves words which can improve the sentiment level. For example, "I'm very happy.", "very" is a word of intensifier. However valenceshifters lexicon involves words which can change the sentiment level. For example, "I'm not happy.", "not" is one of valenceshifters.

4.4.2 Expand dictionary

The original dictionary is in English. We use machine translation to add French and Italian vocabularies. But this time, the French corpus is not available, so we only use English and Italian dictionaries. With DLDA, we can get all sentiment weights of words in corpus. At last, the word which is not included in seed has the same polarity with a seed word if their sentiment weight distance can be ignored.

4.4.3 Rule-based method

We use a scoring strategy to get the sentiment label. In our dictionary, sentiment values are divided into strong pos, strong neg, neutral, weak pos and weak neg. Through DLDA, every word gets a sentiment state. We map the sentiment

state to a number as shown in Table 2. We accumulate a sentence score that reflects the sentiment label as shown in Table 3.

Sentiment state	Word score
weak neg(only)	-1
strong neg(only)	-2
strong pos(only)	2
weak pos(only)	1
neutral	0
Intensifier+weak neg	-2
Intensifier+weak pos	2
valenceshifters	when current sentence score is bigger than 0 and current word is in valenceshifters and the score of current word is less than 0, sentence score = sentence score *(-1), or current sentence score is less than 0 and current word is in valenceshifters and the score of current word is more than 0, sentence score strategy is the same. For any other conditions, we simply accumulate the word score.

Table 2. Scoring strategy

Sentence final score	Label
>0	pos
=0	neutral
<0	neg

Table 3. Mapping relationship of sentence final score to sentiment label.

4.5 Experiments

Before Multiling 2015 published the evaluation results, we only did some original experiments for word vectors in a limited period of time.

We conducted experiments on the word analogy task [41]. Due to the known expectation of the word vector, vector(king) - vector(queen) + vector(man) = vector(woman), we can use this method to evaluate the result of the trained word vectors.

The word analogy dataset is just available in English. We firstly translated this dataset into Italian using Google translation. Because of language limitation, we assessed Italian word vector by word analogy task only for reference.

	English	Italian
Accuracy	20.45%	1.22%
Coverage	100%	98.18%

Table 4. Word Vector Evaluation

The word analogy dataset we obtained contains 19544 word groups, and the English training corpus covers 100% of it, while the Italian one covers 98.18%, as shown in Table 4. As a reference, another model--word2vec's[42] accuracy is between 50%~60% by different parameter settings and training and testing corpus. As we can see that there is a big space for improvement.

4.6 Evaluations

According to the released evaluation results of MMS Task, our results seem bad. Looking back to our submitted summary, we find that most topics in Arabic and Chinese doesn't get enough words (250) for the summary, which we think is caused by that average sentence similarity in the corpus is higher than our specific redundancy similarity threshold of 0.5. Another reason for the bad evaluation result is that our sentence extraction strategy needs more deep semantic features from hLDA and more traditional features like keywords, entities, similarity with title and so on. Although we have a good hLDA modeling tree structure, we can't make full use of the hLDA tree and only choose abstractive feature for hLDA modeling result. Sentence extraction strategy with more features can possibly improve the system.

The OnForumS Task of Multiling 2015 has released the evaluation results about its three subtasks, i.e., Content Linking, Argument Label and Sentiment Label. Although the performance of our linking task is not ideal, we have got the best performance in Argument Label and Sentiment Label.

As to the unsatisfied performance of linking task, there may be three major reasons. First, the training corpus of Word Vectors is collected from Wikipedia by us, which is different from the testing corpus of OnForumS. Second, too many wrong links were selected by the improper threshold of similarity computation. Third, word vectors may sometimes lead to "excessive linked" which means that the word vectors can not only help link two relevant sentences without the same vocabulary but also wrongly link two unrelated sentences by scoring them with much higher semantic similarity. We will work more on these problems in future.

As to Argument Label and Sentiment Label, the following Table 5 shows the precisions of all

systems published by OnForumS. Our systems are the top 2 ones. This has demonstrated the effectiveness of our method. But there are no recall and F-measure results from OnForumS. We think that the reason may be that the evaluation method being used is human judgments.

System No.	Precision (ARGM)	Precision (SENTM)
CIST-RUN1	**0.990601504**	**0.946050096**
CIST-RUN2	**0.988527725**	**0.933837429**
BASE-first	0.974358974	0.927027027
BASE-overlap	0.915531335	0.922077922
UWB-RUN1	0.896153846	0.897435897
JRC-RUN2	0.891891892	0.895752896
USFD_UNITN-run2	0.884848485	0.885714286
USFD_UNITN-run1	0.881578947	0.88030888
JRC-run1	0.859813084	0.874251497

Table 5. Result of Labels

5 Conclusion

Through this MMS task, we find out that we really need to improve our sentence extraction strategy. In future, we will improve the corresponding methods. The information provided by the hLDA tree for different topics, different languages and different knowledge bases should be mined out deeply. In fact, the tree features we have utilized in this MMS task is too little, we need to add more features to improve our summary results. What's more, the sentence similarity for sentence scoring and redundancy removing is also a very important factor for summary length and content.

As we can see, compared with MMS, OnForumS is a new task which involves more natural language technologies besides summarization. This is the first time we tried on it. In future work, we need to study for more and better methods, especially for content linking, argument labeling, sentiment analysis, sentiment labeling and multilingual data. We will also study more about word embedding models and hope to obtain better word vectors via more fitful training experiments.

6 Acknowledgements

This work is supported by the National Natural Science Foundation of China under Grant 71231002, 61202247, 61202248 and 61472046; EU FP7IRSES MobileCloud Project (Grant No. 612212); the 111 Project of China under Grant B08004; Engineering Research Center of Information Networks, Ministry of Education.

Reference

[1] Xiaojun Wan, Jianwu Yang and Jianguo Xiao, 2006. Using Cross-Document Random Walks for Topic-Focused Multi-Document Summarization. WI 2006 Main Conference Proceedings.

[2] John M. Conroy and Judith D. Schlesinger, 2008. CLASSY and TAC 2008 Metrics. TAC 2008 Proceedings.

[3] Shih-Hsiang Lin and Berlin Chen, 2009. THE NTNU SUMMARIZATION SYSTEM AT TAC 2009. TAC 2009 Proceedings.

[4] Shu Gong, Youli Qu and Shengfeng Tian, 2009. Summarization using Wikipedia. TAC 2010 Proceedings.

[5] Renxian Zhang, You Ouyang and Wenjie Li, 2011.Guided Summarization with Aspect Recognition. TAC 2011 Proceedings.

[6] Marina Litvak, Natalia Vanetik, 2013. Multilingual Multi-Document Summarization with POLY. Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization.

[7] Mohamed Abdel Fattah, 2014. A hybrid machine learning model for multi-document Summarization. Appl Intell (2014) 40:592–600 DOI 10.1007/s10489-013-0490-0.

[8] D. M. Blei, A. Y. Ng and M. I. Jordan: Latent Dirichlet allocation. Journal of Machine Learning Research. 3,993-1022 , 2003

[9] Arora Rachit, and Balaraman Ravindran, 2008. Latent dirichlet allocation based multi-document summarization,Proceedings of the second workshop on Analytics for noisy unstructured text data. ACM, 2008.

[10] Krestel Ralf, Peter Fankhauser and Wolfgang Nejdl.2009. Latent dirichlet allocation for tag recommendation. Proceedings of the third ACM conference on Recommender systems. ACM, 2009.

[11] Griffiths T., Steyvers M., Blei D. and Tenenbaum J., 2005. Integrating topics and syntax. Advances in Neural Information Processing Systems 17. L. K. Saul, Y. Weiss, and L. Bottou, eds. MIT Press, Cambridge, MA,2005:537–544.

[12] Blei D. and Lafferty J., 2006. Dynamic topic models. In International Conference on Machine Learning (2006). ACM, New York, NY, USA:113–120.

[13] Wang C. and Blei D., 2009. Decoupling sparsity and smoothness in the discrete hierar-

chical Dirichlet process. Advances in Neural Information Processing Systems 22. Y. Bengio, D. Schuurmans, J. Lafferty, C.

[14] Teh Y., Jordan M., Beal M. and Blei D., 2006. Hierarchical Dirichlet processes. J. Am. Stat. Assoc. 101, 476(2006):1566–1581.

[15] Blei D., Griffiths T. and Jordan M., 2010. The nested Chinese restaurant process and Bayesian nonparametric inference of topic hierarchies. J. ACM 57, 2 (2010):1–30.

[16] Asli Celikyilmaz and Dilek Hakkani-Tur.2010.A hybrid hierarchical model for multi-document summarization. Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 815–824, Uppsala,Sweden,11-16 July 2010.

[17] Pang, B., L. Lee. Opinion mining and sentiment analysis [J]. Foundations and trends in information retrieval, 2008, 2(1-2): 1-135.

[18] Li Shi, Ye Qiang, Li Yijun, R. Law. Mining features of product from Chinese customer online reviews [J]. Journal of Management Sciences in China, 2009, 315 12(2): 142-152

[19] Yoshua Bengio, Rejean Ducharme, Pascal Vincent, and Christian Jauvin. A neural probabilistic language model. Journal of Machine Learning Research (JMLR), 3:1137–1155, 2003.

[20] Mnih, Andriy, and Geoffrey Hinton. "Three new graphical models for statistical language modelling." Proceedings of the 24th international conference on M Mnih, Andriy, and Geoffrey E. Hinton. "A scalable hierarchical distributed language model." Advances in neural information processing systems. 2009.achine learning. ACM, 2007.

[21] Huang, Eric H., et al. "Improving word representations via global context and multiple word prototypes." Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. Association for Computational Linguistics, 2012.

[22] Lai Siwei, Xu Liheng, Chen Yubo, Liu Kang, Zhao Jun "Chinese Word Segment Based on Character Representation Learning." Journal of Chinese Information Processing 27.5 (2013): 8-14

[23] Socher,Richard, et al."Semi-supervised recursive auto encoders for predicting sentiment distributions." Proceedings of the Conference on Empirical Methods in Natural Language Processing.Association for Computational Linguistics,2011.

[24] Turian J, Ratinov L, Bengio Y, et al. A preliminary evaluation of word representations for named-entity recognition[C]//NIPS Workshop on Grammar Induction, Representation of Language and Language Learning. 2009: 1-8.

[25] Y.-J. Tai and H.-Y. Kao: Automatic Domain-Specific Sentiment Lexicon Generation with Label Propagation, iiWAS2013, Vienna,Austria, 2-4 December,2013

[26] M. Speriosu, Nikita, Sudan, Sid Upadhyay, J. Baldridge: Twitter Polarity Classification with Label Propagation over lexical Links and Follower Graph, Proceedings of EMNLP 2011, pages53-63, Edingburgh , Scotland, UK, July 27-31,2011

[27] Q. Z. Mei, X. Ling, M. Wondra, H. Su, C. X. Zhai: Topic Sentiment Mixture: Modeling Facets and Opinions in Weblogs, WWW2007,Banff, Alberta, Canada, May 8-12,2007

[28] Y. Jo, A. Oh: Aspect and Sentiment Unification Model for Online Review Analysis. WSDM'11, Hong Kong, China, February 9-12,2011

[29] C. lin, Y. He, Joint Sentiment/Topic Modelfor Sentiment Analysis:, CIKM'09 ,Hong Kong, China, November 2-6,2009

[30] Mukherjee, B Liu, Aspect Extraction through Semi-Supervised Modeling, ACL'12 Proceddings of the 50th Annual Meeting of the Association for Computational Linguistics, Long Papers-Volume 1,2012

[31] W. X. Zhao, J. Jiang, H. F. Yan, X. M. Li: Jointly Modeling Aspects and Opinions with a MaxEnt-LDA Hybrid, Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing,Pages 56-65,MIT, Massachusetts, USA, October 9-11,2011

[32] S. Matsumoto, H. Takamura, M. Okumura: Sentiment Classificaiton Using Word Subsequences and Dependency Sub-trees, pages 31-311,PAKDD 2005

[33] D. Davidov, O. Tsur, A. Rapport: Enhanced Sentiment Learing Using Twitter Hashtag and Smileys, 23rd Intrenational Conference on Computational Linguistics, Posters Volume, Beijing, China, August 23-27, 2010

[34] M. Q. Hu, B. Liu: Mining and Summarizing Customer Reviews, KDD'04, Seattle, Washington, USA, August 22-25,2004

[35] X. Hu. L. Tang, J. L. Tang, H. Liu: Exploiting Social Relations for Sentiment Analysis in Microblogging, WSDM'13,Rome, Italy, February 4-8,2013

[36] R. Socher, J. Pennington, E. H. Huang, A.Y. Ng, C. D. Manning: Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions.Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2011

[37] Akkaya, J. Wiebe and R. Mihalcea: Subjectivity Word Sense Disambiguation. Proceedings of the Conference on Empirical Mehtods in Natural Language Processing (EMNLP), 2009

[38] Y. Lu, M. Castellanos, U. Dayal and C. X.Zhai: Automatic Construction of a Context-Aware Sentiment Lexicon: An Optimization Approach, WWW2011, March 28-April 1,2011

[39] Xue Chen;Wenqing Tang ; Hao Xu ;Xiaofeng Hu Semantics, Knowledge and Grids (SKG), 2014 10th International Conference on DOI:10.1109/SKG.2014.20 Publication Year: 2014 , Page(s): 49-56

[40] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation[J]. Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014), 2014, 12.

[41] Tomas Mikolov,Kai Chen,Greg Corrado,et al. Efficient estimation of word representations in vec tor space [BE/OL].[2014-09-19].http://arxiv.org/abs/1301.3781v3

[42] Mikolov T, Sutskever I, Chen K, et al. Distri buted Representations of Words and Phrases and th eir Compositionality[J]. Eprint Arxiv, 2013.