NUDT @ CLSciSumm-18

Pancheng Wang , Shasha Li , Ting Wang , Haifang Zhou , Jintao Tang

School of Computer Science, National University of Defense Technology
Changsha, China, 410073
1192869695@qq.com, lishasha198211@163.com, tingwang@nudt.edu.cn
haifang_zhou@163.com, tangjintao@nudt.edu.cn

Abstract. In this paper, we introduce the NUDT system for this year's CL-SciSumm 2018 task at the BIRNDL 2018 Workshop. For task 1a, we identify the related text spans referred to the citation with random forest model, exploring multiple features. Additionally, we integrate random forest model with BM25 and VSM model and apply a voting strategy to select the most related text spans. Besides, we explore the language model with word embeddings and integrate it into the voting system to improve the performance. For task 1b, we use multi-features random forest classifier to identify the facet of the cited sentences.

Keywords: Random Forest Model, Voting System, Word Embeddings.

1 Introduction

The rapid growth of scientific papers and the need for a researcher to move into another brand-new domain generate the demand of scientific summarization. Scientific summarization has been studied for years since (Simone et al ,2002)[1]. And (Qazvinian and Radev,2008)[2] take the citation summary of a reference paper into account to produce a summary of a single scientific article. As time goes on, researchers go further to take advantage of citation-contexts which identify the related text spans in the reference paper correlated with the citations to produce summaries.

The CL-SciSumm18 task can be dated back to the BiomedSumm Track at the Text Analysis Conference 2014, which concentrates on the biomedical dataset. In the next two years, the CL-SciSumm task was held respectively as part of the Joint Workshop on BIRNDL at JCDL and SIGIR.

The CL-SciSumm task of this year is also organized as part of SIGIR2018. On contrast with CL-SciSumm2017, it has increased 10 articles to the training corpus (up to 40 articles) and a new test set of 10 articles is released this year. The task description is as follows:

- Task 1A: for each citance, identify the spans of text (cited text spans) in the reference paper that most accurately reflect the citance. These are of the granularity of a sentence fragment, a full sentence, or several consecutive sentences (no more than 5)

- Task 1B: For each cited text span, identify what facet of the paper it belongs to, from a predefined set of facets
- Task 2: Finally, generate a structured summary of the RP from the cited text spans of the RP. The length of the summary should not exceed 250 words.

In this paper, we describe our methods which are used to solve task 1a and 1b. As for task 1a, we first consider to regard the task as an information retrieval problem and draw on the method of [3]. We extend language model with our pre-trained AAN word embeddings measuring the similarity between words in a query and a document. Besides, we implement the BM25 model and VSM model with TF-IDF weighting the similarity of the citation and the reference contexts. Then, we apply a voting strategy to select the most related text spans. We also explore the supervised classification method to deal with task 1a, using multi-features random forest model to treat the task as a classification problem. As for task 1b, we use another feature-rich classifier to identify the discourse facets, contingent on the system output of task 1a.

2 Related Work

There has been a large number of related works [14,15,16] since the BiomedSumm Track was released.

For the text spans identification according to citations, the methods can be categorized into two classes, classification task and retrieval task. The former methods include [5,6,7,8], the author of [5] used four classifiers with different features to vote for the final result. [6] proposed a method using SVM with features like tf-idf, named entity features and position information of the reference sentence. [7] computed features based on sentence-level and character-level tf-idf scores and word2vec similarity and then used logistic regression to decide sentences to be selected or not. In a sense, [8] also used classification to do task 1a, they integrated the results from several fundamental methods and voted for the results. Retrieval task, or rather ranking task is explored more than classification task when doing task 1a. Based on the traditional semantic similarity, different strategies are applied. [9] created an index of the reference papers and treating each citance as a query and the results were ranked by VSM and BM25 model. [10] used tf-idf and LCS for the syntactic score and pairwise neural network ranking model to calculate semantic relatedness score.

For facet identification, many teams used bag of words methods [5,11]. Other methods include the classification method using an SVM and CNN[12], [9] created an index of cited text and a majority vote was taken to find the facets.

For the task of summary generation, [11] used a similarity score to choose the sentence with top score in the same facet to be added in the summary. [5] used bisecting K-means and MMR to cluster and extract sentences. [8] combined hLDA knowledge for content modeling and using DPPs to enhance the diversity of the summary. [13] trained a linear regression model to learn the scoring function of each sentence.

3 Methods for Task 1A

In this section, we describe the method that we use to identify the related text spans in the reference paper in detail.

3.1 Sentence preprocessing

The official dataset[1] of CL-SciSumm18 comprises 40 annotated sets of citing and referenced papers in the training set and 10 in the test set. Since the papers are transformed from PDF format to XML or TXT format, there exists a bunch of format mistakes and futile characters in the dataset. Hence, it's essential to preprocess the sentences in the dataset before we set out to deal with the task.

- Sentence processing: we use NLTK to tag the part of speech and move punctuations and stop words.
- Sentence filtering: based on the former step, we try to filtrate sentences which have apparently more unreasonable characters than those that are more likely to be candidate of the retrieved sentences. To find out the error threshold, we establish an English word dictionary composed of 103976 English words[2] and view it as a judger to determine a word legal or not. We count the ratio of illegal words in each of the 566 cited sentences of reference papers in the training set and choose the error ratio threshold as 0.4, which means sentences comprise 40% or more illegal words will be filtrated at the very beginning. Our statistics result is showed below:

Table 1. the error ratio of illegal words in cited sentences of reference papers in the training set

Error ratio	number
0 – 10%	309
10% - 20%	160
20% - 30%	66
30% - 40%	27
> 40%	4

3.2 Information Retrieval Model with Word Embeddings

[3] puts forward methods that extends language models for information retrieval by incorporating word embeddings and domain ontology to address shortcoming of LM for identification of relevant text spans given a citation text.

[1] https://github.com/WING-NUS/scisumm-corpus
[2] https://download.csdn.net/download/sxtuwy/9824178

In information retrieval model, we treat task 1a as a retrieval problem and refer to the citation as query and reference text spans as documents, so we return a list of sentences as candidates according to the query. The original model in [3] is:

$$p(q_i \mid d) = \frac{f_{sem}(q_i, d) + \mu p(q_i \mid C)}{\sum_{w \in V} f_{sem}(w, d) + \mu} \tag{1}$$

The model is an improved LM that using Dirichlet smoothing and the cosine similarity of word pairs based on word embeddings taking the place of word frequencies. Where f_{sem} is a function to measure semantic relatedness of the query term q_i to the document d, C is the entire smoothing corpus, V is the vocabulary of C and μ the Dirichlet smoothing parameter.

f_{sem} is defined as below:

$$f_{sem}(q_i, d) = \sum_{d_j \in d} s(q_i, d_j) \tag{2}$$

Where：

$$s(q_i, d_j) = \begin{cases} \phi(e(q_i).e(d_j)) \ , (e(q_i).e(d_j)) > \tau \\ 0 \qquad\qquad otherwise \end{cases} \tag{3}$$

Here the transformation (ϕ) of dot products between the word embeddings representation of query word q_i and document word d is a logit function:

$$\phi(x) = \log(\frac{x}{1-x}) \tag{4}$$

As for τ, the value is set to be two standard deviations larger than the average value of cosine of embeddings.

We borrow ideas from the above model and present our two improved strategy.

First, we train our own word embeddings according to the AAN(ACL Anthology Network) corpus[4][3]. Since CL-SciSumm18 dataset consists of papers from ACL Anthology corpus, it's reasonable to train specific embeddings concentrated on the CL fields. The AAN corpus include 22486 CL papers, we first use the same preprocessing strategy as described above and then use word2vec tool from gensim to train our own word embeddings[4].

---

[3] http://clair.eecs.umich.edu/aan/index.php

[4] The embeddings are trained with the setting of vector size 400, negative sampling, windows size of 5, minimum count of 5.

To validate the effectiveness of our embeddings, we also download GoogleNews embeddings and use the language model I realized according to the idea above to compare the performance of the two embeddings. Table 2 shows that our AAN embeddings performs much better than GoogleNews ones based on the test-set 2017

Table 2. performance of AAN and GoogleNews embeddings

	Precision@5	Recall@5	Micro-F1 @5
AAN embeddings	0.059	0.202	0.0913
GoogleNews embeddings	0.032	0.108	0.0475

Second, we try to improve the performance of the language model with the section information taking into account heuristically. To validate the feasibility of the idea, we separate a reference paper by sections and apply LDA(Latent Dirichlet Allocation) and LSI(Latent Semantic Index) model to calculate the cosine value between a citation and one section respectively. We carry out the experiment on test-set 2017 and separately compute the ratio whether the section that the reference sentences locate in is in the top 2 or top 3 most similar section according to the LDA and LSI value between section and citations . Our results show that our idea with section taking part in is feasible and LSI model has the upper hand against LDA model.

Based on the above experiment, we modify the language model of (1) by adding section similarity:

$$p(q_i \mid d) = \frac{f_{sem}(q_i, d) + \mu p(q_i \mid C)}{\sum_{w \in V} f_{sem}(w, d) + \mu} * \text{cosine}_{q_i \in q, d \in \text{section}}(LSI[q], LSI[\text{section}]) \quad (5)$$

Compared to the former model, we integrate the cosine value of query and section in LSI space to calculate the probability of query word q_i when given a document d .Here, $LSI[q]$ means the topic distribution of query q ,where the topic number is 50. $LSI[\text{section}]$ means the topic distribution of the given section and the topic number is the same.

3.3 BM25 and VSM model

In addition to the language model with word embeddings, we also implement BM25 and VSM(Vector Space Model) model, since the two models are classical retrieval model and may serve as baselines during my experiment.

– BM25: the model is defined as follow

$$f(q,d) = \sum_{w \in q \cap d} c(w,q) \frac{(k+1)c(w,d)}{c(w,d) + k(1-b+b\frac{|d|}{avedl})} \log(\frac{N - n_w + 0.5}{n_w + 0.5}) \quad (6)$$

Where q, d denotes query and document respectively, $c(w,q)$ denotes the frequency that word w appears in q. $c(w,d)$ denotes the frequency that word w appears in document d. $|d|$ is the length of document d. $avedl$ is the average length of all the documents. N is the number of the documents and n_w means the number of documents that word w appears in.

Besides, k and d are hyperparameters and the values are 1.25 and 0.75 respectively, according to our experience.

- VSM: vector space model is another popular model to be applied in retrieval field. We use TF-IDF(term frequency and inverse document frequency) value to constitute the vector space.

3.4 Random Forest Classifier

Our preceding meta-models are all unsupervised models which make full use of the sematic and lexical relevance between citations and reference papers. Since CL-Scisumm dataset has manual annotation in training sets, supervised approach can be a good solution to task 1a.

We apply random forest model to solve the problem, the following features are chosen:

- Jaccard similarity: the quotient of the intersection divided by the union between the citation and the candidate reference sentence.
- BM25 similarity: the BM25 similarity value between the citation and the candidate reference sentence as we described before.
- Vectorized TF-IDF similarity: the cosine value between the citation and the candidate reference sentence which are represented by TF-IDF value in vector space.
- Section similarity: the cosine value between the citation and the section that the candidate reference sentence locates in via LSI model.
- AAN word embeddings alignment: the value is defined as follow:

$$f(citation, sentence) = \sum_{c_i \in citation} \sum_{s_j \in sentence} \frac{f(c_i, s_j)}{|sentence|} \tag{7}$$

Where $f(c_i, s_j)$ is the same as our former definition in (3).

- Average distance of AAN word embeddings: we add up all the word embeddings in the citation and the candidate reference sentence respectively, normalize the vectors and get the cosine value as the average distance.

Because of the extreme imbalance of the labels of the data, we consider oversampling strategy to deal with this situation. Here we apply SMOTE+ENN technique to increase the number of label 1.

3.5 Voting Method

Based on the preceding models we establish, we consider using voting method to integrate the results of the models.

Here we apply two layers of voting to select sentences as the final candidate output. The mechanism is showed below.

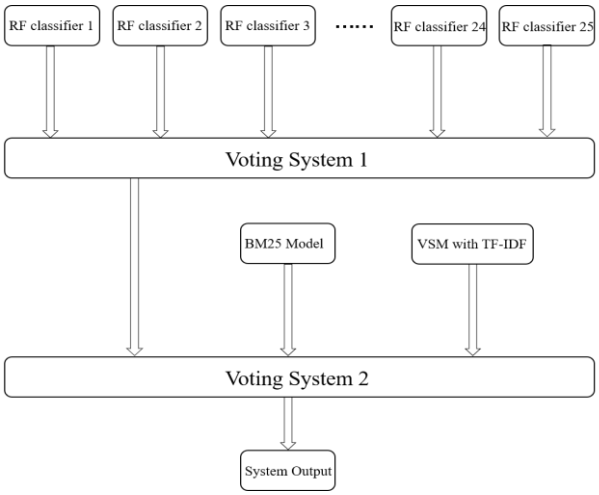

Fig. 1. The framework of the voting system for task 1a

Since our oversampling strategy SMOTE+ENN will produce a number of positive samples in every run, the performance of the random forest is closely connected with the new samples. Hence, we consider saving the random forest models which perform well on the Test-Set 2017.

We save 50 RF models that perform well individually at first, then we determine the number of models and the threshold of voting to be 25 and 17 respectively according to Fig.2 and Table 3 in the first voting layer.

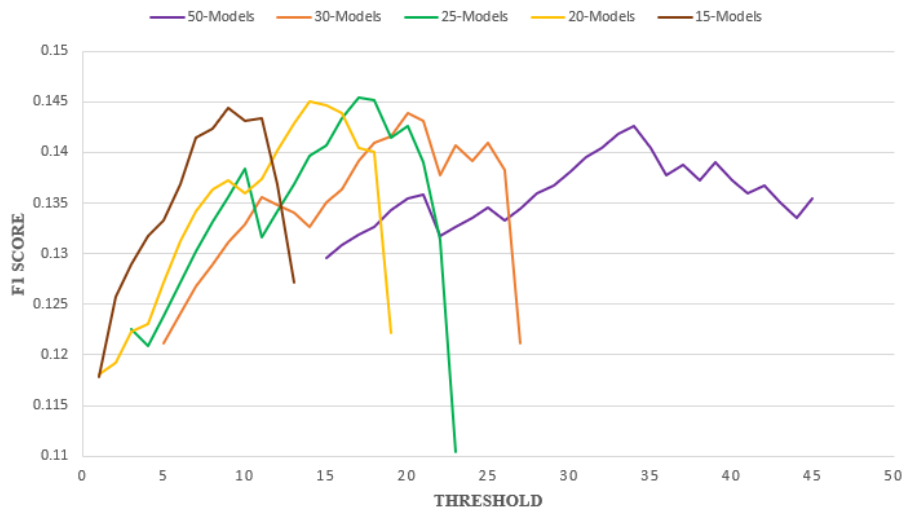

Fig. 2. The performance of RF models with different numbers and threshold when voting

Table 3. the best threshold for different models

Number of models	Best threshold	Micro-Precision	Micro-Recall	Micro-F1
50	34	0.118	0.179	0.1426
30	20	0.120	0.179	0.1436
25	17	0.122	0.179	**0.1453**
20	14	0.121	0.179	0.1451
15	9	0.1209	0.179	0.1444

In the second voting layer, we integrate the output of voting layer 1, the top ten sentences of BM25 and the top ten sentences of VSM model to vote for the ultimate results. Only the sentences that are included in all three models will be chosen as the output sentences.

Besides, we do pruning and padding operation on the results. In case the corresponding output of a citation is nonexistent, then we return the top 2 sentences in BM25 model as the output. In case the corresponding output of a citation is more than 4 sentences, then we return the top 4 sentences in BM25 model as the output.

In addition to the above voting system, we also using another voting shown in Fig.3 and also submit a system.

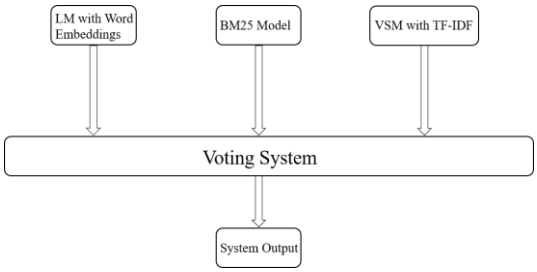

Fig. 3. The framework of another voting system

4 Method for task 1b

For task 1b, we need to identify what facet of the paper the selected sentences belong to. And in this section, we are going to describe our method applied for task 1b.

We train 4 random forest classifiers for the facet Method, Aim, Implication and Result respectively. It's worth mentioning that we ignore training classifier for the facet Hypothesis, owing to the fact that there is little samples of Hypothesis in the dataset.

The features for the four classifiers are the same.

We establish four Bag of Words model for the four facets separately, then we use the BOW representation to calculate scores of each sentence and we get the similarities between one sentence and the four facets as four features.

Another features we use are as follows:

- Number of numeric character: we count the number of numeric characters for each input sentence as a feature.
- Relative position in the section: relative position for one sentence in the section which the sentence is located.
- Relative position in the full paper: relative position for one sentence in the full reference paper.

We train the models on the training-set 2017 and apply the following strategies to get the final forecast output. If the probability of the positive label from one classifier is over 0.5, then we return the facet correlated with the classifier. In case none of the probabilities is over 0.5, if none is over 0.2, then we identify the facet as Hypothesis. Otherwise, we identify the facet as Method.

5 Experiment Results

For task 1a, we submit 4 systems and the settings are as follows:

- System 1: a voting system combined 20 random forest models with the voting threshold to be 14

- System 2: a voting system combined 25 random forest models with the voting threshold to be 17
- System 3: a two-layer voting system shown in Fig 3.
- System 4: a two-layer voting system shown in Fig 1.

We evaluate our systems using micro average metric on test-set 2017, which is part of training-set 2018.

The results are shown in table 4.

Table 4. results of the four systems on test-set 2017

System id	Precision	Recall	F1-score
1	0.1127	0.1703	0.1357
2	0.1127	0.1703	0.1357
3	0.1116	0.2052	0.1446
4	0.1309	0.1703	**0.1480**

From the above table, we could see that both the voting system consisting of random forest models and the two-layer voting system achieve high performance on test-set 2017, which proves the validity of our methods.

Table 5. distribution of facets on test-set 2017

Facets	Method	Result	Aim	Implication	Hypothesis
number	143	11	1	0	0

As for task 1b, because of the severe imbalance of the dataset shown in table 5. The test-set 2017 has 155 sentences in total, but 92.25% of the facets are methods, and result accounts for 7.1%. On the contrast, the facet implication and hypothesis do not appear in the dataset. We consider not evaluating the performance of identification of facets but adjust the parameters of the models according to the performance on the training set.

6 Conclusion

This paper has focused on our methods applied for task 1a and 1b of the CL-SciSumm 2018. For task 1a, we find the baseline BM25 model can almost achieve the best performance on test-set 2017. Although the robustness of the result is not so convincing, but the phenomenon indicates that semantic-based citation identification is the main stream of the former exploration and the popular deep learning methods do not achieve satisfactory results because of the limitation of the scale of the dataset. We also get that the voting method is an effective strategy to improve the performance of the systems. For task 1b, a valuable and heuristic conclusion is that the distribution of facets to the reference sentences according to the citations is imbalanced and the summary merely extracted from the cited spans may not comprehensive and com-

plete. Hence, how to combine citation information and other useful information for summary generation could be a consideration when doing scientific summarization.

References

1. Simone Teufel M M. Summarizing Scientific Articles - Experiments with Relevance and Rhetorical Status[C]. Computational Linguistics. 2002:2002.
2. Qazvinian V, Radev D R. Scientific paper summarization using citation summary networks[C]. International Conference on Computational Linguistics. Association for Computational Linguistics, 2008:689-696.
3. Cohan A, Goharian N. Contextualizing Citations for Scientific Summarization using Word Embeddings and Domain Knowledge[J]. 2017:1133-1136.
4. Radev, D.R., Muthukrishnan, P., Qazvinian, V.: The acl anthology network corpus. In: Proceedings of the 2009 Workshop on Text and Citation Analysis for Scholarly Digital Libraries. pp. 54–61. Association for Computational Linguistics (2009)
5. Ma, S., Xu, J., Wang, J., Zhang, C.: NJUST@CLSciSumm-17. In: Proc. of the 2nd Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL2017). Tokyo, Japan (August 2017)
6. Cao, Z., Li, W., Wu, D.: Polyu at cl-scisumm 2016. In:BIRNDL 2016 Joint Workshop on Bibliometric-enhanced Information Retrieval and NLP for Digital Libraries (2016)
7. Zhang, D.: PKU @ CLSciSumm-17: Citation Contextualization. In: Proc. of the 2nd Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL2017). Tokyo, Japan (August 2017)
8. Li, L., Zhang, Y., Mao, L., Chi, J., Chen, M., Huang, Z.: CIST@CLSciSumm-17: Multiple Features Based Citation Linkage, Classification and Summarization. In: Proc. of the 2nd Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL2017). Tokyo, Japan (August 2017)
9. Felber, T., Kern, R.: Query Generation Strategies for CL-SciSumm 2017 Shared Task. In: Proc. of the 2nd Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL2017). Tokyo, Japan (August 2017)
10. Prasad, A.: WING-NUS at CL-SciSumm 2017: Learning from Syntactic and Semantic Similarity for Citation Contextualization. In: Proc. of the 2nd Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL2017). Tokyo, Japan (August 2017)
11. Dipankar Das, S.M., Pramanick, A.: Employing Word Vectors for Identifying,Classifying and Summarizing Scientific Documents. In: Proc. of the 2nd Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL2017). Tokyo, Japan (August 2017)
12. Lauscher, A., Glavas, G., Eckert, K.: Citation-Based Summarization of Scientific Articles Using Semantic Textual Similarity. In: Proc. of the 2nd Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL2017). Tokyo, Japan (August 2017)
13. Abura'Ed, A., Chiruzzo, L., Saggion, H., Accuosto, P., lex Bravo: LaSTUS/TALN @ CL-SciSumm-17: Cross-document Sentence Matching and Scientific Text Sum- marization Systems. In: Proc. of the 2nd JointWorkshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL2017). Tokyo, Japan (August 2017)

14. Kokil Jaidka, Muthu Kumar Chandrasekaran, Devanshu Jain, and Min-Yen Kan (2017). Overview of the CL-SciSumm 2017 Shared Task, In Proceedings of the Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL 2017), Tokyo, Japan, CEUR.
15. Jaidka, K., Chandrasekaran, M. K., Jain, D., & Kan, M. Y. (2017). The CL-SciSumm shared task 2017: results and key insights. In Proceedings of the Computational Linguistics Scientific Summarization Shared Task (CL-SciSumm 2017), organized as a part of the 2nd Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries (BIRNDL 2017).
16. Jaidka, K., Chandrasekaran, M. K., Rustagi, S., & Kan, M. Y. (2017). Insights from CL-SciSumm 2016: the faceted scientific document summarization Shared Task. International Journal on Digital Libraries, 1-9.Jaidka, K., Chandrasekaran, M. K., Rustagi, S., & Kan, M. Y. (2017). Insights from CL-SciSumm 2016: the faceted scientific document summarization Shared Task. International Journal on Digital Libraries, 1-9.