A Text Summarization Method Based on Semantic Similarity Among Sentences

Yu-bing HOU¹,a,*

¹China Information Technology Designing & Consulting Institute Co., Ltd., Shanghai, China
³houyb@dimpt.com
*Corresponding author

Keywords: Automatic text summarization, Graph-based ranking, Graph model, Semantic similarity.

Abstract. In recent years, more and more attention has been paid to the graph-based text summarization. In terms of this method, the document is transformed into a text graph and then sentences are ranked by taking into account the global information of the document. However, the similarity measure between sentences is mostly limited to word layer in previous studies, which makes it difficult to measure the semantic similarity accurately; besides, the features of sentence has also been disregarded. In view of the defects mentioned above, this paper proposes a semantic-word two layer similarly measure model and defines topic relativity as well as position sensitivity, so as to optimize the sentence ranking results. Through the evaluation on DUC datasets, performance of this approach shows greater improvement compared with a number of baseline systems.

Introduction

Automatic text summarization has a long history dated back to 1960s, despite of which up to now, extraction-based summarization that extracts sentences directly from the original documents so as to generate summaries is still the mainstream method in this research area. Against this background, sentences ranking is surely the core issue. In recent years, graph-based summarization has aroused more and more attention, because according to this method, sentences are ranked by taking into account the similarity between sentences as well as iterative calculating through the global rather than the local information of the document. Therefore, the main idea behind this method is familiar with the artificial text summarization. In graph-based summarization, document is transformed into a graph in which nodes represent the text units (terms or sentences), whereas edges weigh the similarity between pairs of nodes. After that, graph-based ranking algorithms such as PageRank [1] or HITS[2] are used to rank the nodes. Based on this framework, the node which is closely connected with many others nodes is considered more likely to carry the core information and will have higher weights after ranking.

In graph-based ranking algorithm, the similarity measurement precision between nodes could affect the ranking results to a great degree. Therefore, in graph-based summarization, the similarity measure between text units is the most important thing. Most of the previous studies choose the sentences as the graph nodes while at the same time the similarity measure between sentences is mostly limited to word layer, such as the use of word co-occurrence [3], cosine similarity[4-6] or WordNet to measure the word similarity[7]. However, it is difficult for the word layer measurement to measure the sentences semantic similarity accurately.

On the other hand, though the graph-based summarization has taken into account the similarity between sentences, there is also a major factor which has been neglected: the correlation between sentence and document topic. This may cause local optimum in sentence ranking. For example, there is a large section of the non-core content in an article, but sentences in this section are closely related to each other. As a result, after sentence ranking, high weight sentence may appear in this section, but this local optimum sentence cannot represent the main idea of the whole article.

Besides, the graph-based summarization also overlooks the features of text units, such as sentence length, sentence position. In many documents, especially in news, the main ideas are always shown in the first paragraph, for which the overlook of sentence position will surely affect the ranking result.
Focusing on those problems, this paper has the main contributions summarized as follows:

- A two-layer similarity measure model that combines the LDA topic model [8] with cosine similarity is put forward, with the capacity to measure the similarity between sentences in both the word layer and semantic layer.
- The definitions of topic relativity and position sensitivity are proposed to optimize the sentence ranking result. In order to utilize the two features, Biased-PageRank [9] algorithm is chosen to rank sentences.

**Two-Layer Similarity Measure Model**

**Semantic Layer**

In LDA topic model, the probability distribution of topics is used to represent the documents. In our approach, each sentence has a topic probability distribution, and with more similarities, the two topic distributions would have higher sentence semantic similarity.

Document-topic distribution and topic-word distribution can be obtained after the training of the LDA topic model. We use \( P(W|T) \) to represent the topic-word distribution and \( P(T|D) \) to represent document-topic distribution. Therefore, \( P(W_i|T_j) \) means the probability that word \( W_i \) represents topic \( T_j \) while \( P(T_i|D_k) \) means the probability that document \( D_k \) belongs to topic \( T_j \). The dimension of \( W \) is the word count of corpus, the dimension of \( T \) is the topic numbers of LDA topic model, and the dimension of \( D \) is the number of documents in this corpus. In a document, the sentences topic probability distribution that we are looking for is \( P(T|S) \); for a sentence \( S_r \) in document \( D_k \), the probability that it belongs to topic \( T_j \) is given by Bayesian formula:

\[
P(T_j|S_r) = \frac{P(S_r|T_j)P(T_j)}{P(S_r)}
\]

(1)

\( P(S_r|T_j) \) is the probability that sentence \( S_r \) represents topic \( T_j \), \( P(T_j) \) is the probability that the corpus belongs to topic \( T_j \), while \( P(S_r) \) is the prior probability of sentence \( S_r \).

Sentence is made up of words, for which topic-sentence probability distribution \( P(S|T) \) can be calculated on the basis of topic-word probability distribution \( P(W|T) \). The calculation of \( P(S_r|T_j) \) has been discussed in detail in [10]:

\[
P(S_r|T_j) = P(T_j|D_k) \ast P(D_k) \ast \sum_{W_i \in S_r} P(W_i|T_j)
\]

(2)

Moreover, the \( P(T_j) \) can be calculated as:

\[
P(T_j) = \sum_{k=1}^{M} P(T_j|D_k) \ast P(D_k)
\]

(3)

\( M \) is the number of documents. For \( P(S_r) \) and \( P(D_k) \), we suppose all the sentences and documents have the same prior probability. Hence, with influence in the calculation, \( P(S_r) \) and \( P(D_k) \) can be neglected. Then, through the combination of formula (1) (2) (3), the final formula of sentence-topic probability distribution can be got:

\[
P(T_j|S_r) = \sum_{W_i \in S_r} P(W_i|T_j) \ast P(T_j|D_k) \ast \sum_{k=1}^{M} P(T_j|D_k)
\]

(4)

One thing to note is that for \( j \in \text{Topic Numbers} \), the sum of \( P(T_j|S_r) \) is not equal to 1, so \( P(T|S_r) \), the calculation result of formula (4) cannot be directly regarded as the topic distribution of sentence \( S_r \). In this way, normalization processing is necessary:
For sentence $S_r$ in document $D_k$, we can get the final $P(T|S_r)$ by combining formula (4) with (5).

As an effective method to measure the similarity between two probability distributions, Kullback–Leibler divergence, however, is asymmetric and its value is not in range $[0, 1]$. Therefore, in case of using KL divergence, the result value would be inconvenient to use. Instead, we choose Jensen–Shannon divergence based on the Kullback–Leibler divergence with some useful differences, including symmetry and finite value in range $[0, 1]$. For sentences $P$, $Q$ and their topic probability distribution $P$, $Q$, we can derive their JS divergence as:

$$JS_{PQ} = \frac{1}{2} KL(P \parallel M) + \frac{1}{2} KL(Q \parallel M)$$

(6)

$$M = \frac{1}{2} (P + Q), \text{ KL}(P \parallel M) \text{ is the KL divergence between distribution } P \text{ and } M.$$  

$$KL(P \parallel M) = \sum_i P(i) \ln \frac{P(i)}{M(i)}$$

(7)

Similar to KL divergence, with smaller JS divergence value, there would be higher sentence semantic similarity. Therefore, we use the difference between JS divergence and 1 as the quantized value of sentences semantic similarity. For sentences $P$, $Q$, we can derive their semantic similarity as:

$$SemSim_{PQ} = 1 - JS_{PQ}$$

(8)

**Word Layer**

In order to measure the similarity between sentences in word layer, we transform each sentence of a document to an N-dimensional vector, after which cosine similarity between sentences can thus be calculated. $N$ is the number of words occurring in the document and for each word that occurs in a sentence, its TF*IDF value will be set to the corresponding dimension of this vector. Hence, for sentences $P$, $Q$, their cosine similarity can be calculated as:

$$CosSim_{PQ} = \frac{\sum_{w \in P \cap Q} TF_{w,P} \cdot TF_{w,Q} \cdot IDF(w)^2}{\sqrt{\sum_{w \in P}(TF_{w,P} \cdot IDF(w))^2} \cdot \sqrt{\sum_{w \in Q}(TF_{w,Q} \cdot IDF(w))^2}}$$

(9)

$$TF_{w,P} \text{ is the word frequency of word } w \text{ in sentence } P.$$  

$$TF_{w,P} = \frac{N_{w,P}}{N_P}$$

(10)

$N_{w,P}$ refers to the word occurrence of word $w$ in sentence $P$, and $N_P$ indicates the number of words in sentence $P$.

$IDF_w$ is the inverse document frequency of word $w$:

$$IDF_w = \log\frac{N}{N_w}$$

(11)

$N$ is the total number of documents in corpus, and $N_w$ is the word occurrence of word $w$ in corpus.

For sentences $P$, $Q$, we can derive their similarity by combining formula (8) with (9):
is used to adjust the proportion of $\text{SemSim}$ and $\text{CosSim}$, with its value in range $[0, 1]$. 

**Improvement in Sentence Ranking**

With the adoption of sentences as the graph nodes, a text graph has been constructed, while weighted edges have been generated according to their similarity calculated in Section 3. For this graph, we use weighted PageRank algorithm to rank the sentences:

$$ R(P) = \frac{(1-d)}{N} + d \sum_{Q \in \text{adj}(P)} \frac{\text{Sim}_{PQ}}{\sum_{Q \in \text{adj}(Q)} \text{Sim}_{QQ}} R(Q) $$  

$(13)$

$R(P)$ represents the weight of node $P$, $d$ is the damping factor usually set to 0.85, $N$ is the total number of graph nodes, and $Q \in \text{adj}(P)$ represents all adjacent nodes to $P$. In our weighted undirected graph model, all nodes are coadjacent to each other. In the weighted PageRank, a node contributes more to its similar nodes in weight propagation.

After sentence ranking, we can select the TOP-N sentences and organize them as the original order so as to generate the summary.

As mentioned above, in graph-based summarization, sentences are ranked according to the document global information but the correlation between sentence and document topic as well as the own features of sentences has been neglected. This may lead to inaccurate sentences ranking result. Therefore, we propose the notion of topic relativity and position sensitivity to optimize the sentence ranking.

**Topic Relativity**

In order to measure the topic relativity, we need to mine the document topic and find the right way to represent the topic. In previous studies, the keywords are usually used to represent the topic. For example, in [11] the words with higher TF*IDF value are deemed to be more representative, so its topic relativity of sentence is defined as the quotient between sum of words’ TF*IDF values and the sentence length. Besides, an improved method is proposed in [12]; compared with [11], the values of words are set to 1 or 0 according to a threshold instead of exact TF*IDF values.

However, there has been limited effect in terms of using words to represent the topic despite of the easiness. On account of using LDA topic model, we choose to use the document topic probability distribution to represent the topic, and in Section 3.1, we have obtained the sentence topic probability distribution. So we can calculate the sentence topic relativity in semantic layer by measuring the two probability distributions.

Besides, the formula (8) is used to calculate the similarity between document topic probability distribution and sentence topic probability distribution. For document distribution $D$ of document $D$ and sentence distribution $P$ of sentence $P$, the normalized topic relativity of $P$ can be calculated as:

$$ TR_P = \frac{\text{SemSim}_{PP}}{\sum_{P \in D} \text{SemSim}_{PP}} $$  

$(14)$

In PageRank algorithm, the initial weight of node is useless, but Biased-PageRank allows the use of the initial static weight to get a more reasonable result. According to its topic relativity, a static weight has been provided with the node; through the combination with the formula (13), the new iterative formula is shown as follows:

$$ R(P) = d \times TR_P + (1-d) \sum_{Q \in \text{adj}(P)} \frac{\text{Sim}_{PQ}}{\sum_{Q \in \text{adj}(Q)} \text{Sim}_{QQ}} R(Q) $$  

$(15)$

The random walk of this formula can be explained as follows: with probability $d$, the random walk visits a sentence with a probability proportional to its relativity to the topic; with probability $(1-d)$, the random walk chooses a sentence that is a neighbor of the current sentence with a probability proportional to the link weight in the graph.
Position Sensitivity

Generally speaking, a document often discloses its main idea in the first paragraph, which is especially obvious in news. Therefore, we propose the notion of position sensitivity. For sentence $P$ in document $D$, with more front sentence $P$, weight shall be heavier. Hence, we define the calculation of position sensitivity as follows:

$$PS_P = \frac{1}{\sum_{i=1}^{\text{len}(D)} i^\text{pos}}$$ (16)

$pos$ is the position order of sentence $P$ in document $D$. For example, if $P$ is the second sentence, then $pos=2$, $\text{len}(D)$ represents the sentence number in document $D$.

Combining position sensitivity with formula (15), we obtain the final iterative formula:

$$R(P) = d \ast \frac{(TR_P + PS_P)}{2} + (1 - d) \sum_{Q \in \text{adj}(P)} \frac{\text{Sim}_{PQ}}{\sum_{Q \in \text{adj}(Q)} \text{Sim}_{Q}} R(Q)$$ (17)

Experiment

Experiment Data and Evaluation

The document set of DUC (Document Understanding Conference) is a commonly used dataset to evaluate automatic summarization systems. The task of each session is different. In accordance with our goal, DUC01 and DUC02 focus on single document summarization. Hence, we choose to use DUC02 datasets to evaluate our system. DUC02 datasets includes a total of 596 documents in 59 topics, with the average topic including about 10 documents. Compared with DUC01, DUC02 provides the results of other participant systems, thus making itself a rather ideal dataset.

ROUGE[13] is the official evaluation tool of DUC. By comparing the artificial summaries and system summaries, it can calculate out multi scores: the average recall, precision and F-measure score of ROUGE-1, ROUGE-2, ROUGE-3, ROUGE-4, ROUGE-W, ROUGE-S, ROUGE-L and ROUGE-SU. Recall is the most appropriate evaluation index for summary of fixed length. Our system focuses on 100-word single document summarization, which we choose the recall scores to evaluate our system.

Parameter Tuning

We use the DUC02 datasets as the corpus and use Gibbs sampling to estimate the document-topic and topic-word probability distribution of LDA topic model. The initial parameters of LDA is set as follows: $\alpha = \frac{50}{K}, \beta = 0.01$, and $K$ is the topic number. This parameter setting is commonly used for the actual fast convergence.

Besides, to guarantee the convergence, we set the Gibbs sampling iterations as 10000, BURN_IN as 2000 and THIN_INTERVAL as 100. It means that since the start of the 2000th iteration of Gibbs sampling, we record each sample state every 100 iterations, and the result probability distribution is the average value of all the records. The setting of $K$ is from 10, 20, 30 and until 200, intending to find the best topic number for DUC02 corpus.
Figure 1. ROUGE scores under different topics K. Figure 2. ROUGE scores under different combination.

**Figure 1** is the evaluation result that only uses the semantic measurement (formula (8)) to measure sentence similarity with topic number $K$ on the horizontal and ROUGE recall score on the vertical. The ROUGE-4 score is too low and has little difference within each parameter setting; besides, ROUGE-S, ROUGE-W and ROUGE-SU scores are similar with ROUGE-2 in our experiments, for which we choose to display the remaining 4 scores. It is observed that all scores would reach the peak under $K=20$. Therefore, for DUC02 dataset, we regard the best topic number as 20. One thing to note is that we find that the curvilinear trends of various ROUGE scores are basically similar to each other. Therefore, in the follow-on experiments, ROUGE-2 recall score has been designated to evaluate the system performance.

**Figure 2** shows the ROUGE-2 recall score under different combinations of four characteristics: ‘LDA’ represents the use of JS divergence, ‘CosSim’ represents the use of cosine similarity, ‘Topic’ represents the combination of topic relativity, and ‘Pos’ represents the combination of position sensitivity. Moreover, the horizontal axis is $\lambda$ of formula (12), and the vertical axis is ROUGE-2 recall score. The far left point of each curve is the evaluation result that only uses the semantic measurement, while at the same time the far right point is the evaluation result that only uses the cosine similarity.

We can see that LDA+CosSim+Topic+Pos, the approach combining semantic similarity, cosine similarity, topic relativity and position sensitivity performs much better than other combinations. Due to that the score reaches the peak under $\lambda = 0.2$, we designate this peak score as the final score of our system.

**Experimental Comparison**

We compare ROUGE-2 recall scores with other summarization systems: the numbers of the best three systems in DUC02 are 21, 27 and 28, respectively. Besides, we setup three control experiments as the baselines: (1) randomly select sentences (2) select first sentences that satisfy 100 words (3) use cosine similarity only to measure sentences similarity.

As shown in Table 1, the effectiveness of our two-layer similarity measure model has been actually proved compared with the ‘Random’ approach. However, using ‘LDA’ or ‘LDA+CosSim’ performs worse than ‘CosSim’, which may be due to that the inadequate numbers of documents in DUC datasets weaken the representativeness of training result of LDA topic model. However, LDA+CosSim+Topic, with the combination of semantic, word and topic information shows good performance as expected.

The performance of LDA+CosSim+Topic+Pos is better than the best system of DUC02, for the reason that the four characteristics (semantic similarity, cosine similarity, topic relativity and position sensitivity) have covered most factors that could affect the sentence importance measure.

It is shown that the combination of the position sensitivity leads to a significant improvement, which is because the documents in DUC02 belongs to news types and position is an important indicator for sentence importance measure in news.
Table 1. Performance comparing to baseline and DUC02 TOP-3.

| Systems/Combinations/Baselines | ROUGE-2 recall score |
|--------------------------------|----------------------|
| LDA+CosSim+Topic+Position      | 0.22776              |
| 27                             | 0.22674              |
| 28                             | 0.22550              |
| 21                             | 0.22113              |
| LDA+CosSim+Topic               | 0.20143              |
| First Sentences                | 0.19753              |
| CosSim+Topic                   | 0.19605              |
| LDA+Topic                      | 0.19338              |
| CosSim                         | 0.19326              |
| LDA+CosSim                     | 0.18814              |
| LDA Feature                    | 0.17870              |
| Random                         | 0.12731              |

**Conclusion**

This paper focuses on graph-based single document summarization. For text graph construction, a two-layer sentence similarity measure model has been put forward. On the basis of LDA topic model, we calculate out the sentence topic probability distribution so as to measure the sentence similarity on the semantic layer, while at the same time using cosine similarity to measure sentence similarity on the word layer. Besides, focusing on the defects of previous graph-based summarization, we propose the topic relativity and position sensitivity to optimize the sentence ranking. Furthermore, a modified Biased-PageRank is used to rank the sentences.

The DUC02 dataset has been used to perform qualitative evaluation, and experimental comparison shows that the proposed model outperforms other related work.

**References**

[1] S Gupta, A Nenkova, D Jurafsky. Recent automatic text summarization techniques: a survey[J]. Artificial Intelligence Review, 2017, 47(1):1-66.

[2] YA Al-Khassawneh, N Salim, AI Obasae. Sentence Similarity Techniques for Automatic Text Summarization[J]. Journal of Soft Computing and Decision Support System, 2016, 3(3):35-41.

[3] Goyal, P. L. Behera, and T.M. McGinnity. A Context-Based Word Indexing Model for Document Summarization[J]. IEEE Transactions on Knowledge and Data Engineering, 2013, 25(8):1693-1705.

[4] Erkan, G and D.R. Radev. LexRank: Graph-based lexical centrality as salience in text summarization[J]. J. Artif. Intell. Res.(JAIR), 2004, 22(1):457-479.

[5] Erkan, G and D.R. Radev. LexPageRank: Prestige in Multi-Document Text Summarization[C]. EMNLP, 2004:365-371.

[6] Mihalcea, R and P. Tarau. TextRank: Bringing order into texts[J]. Association for Computational Linguistics, 2004:1-360.

[7] Ferreira, R. A Four Dimension Graph Model for Automatic Text Summarization[C]. 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT), 2013:389-396.

[8] Blei, D. M., A.Y. Ng, and M.I. Jordan. Latent dirichlet allocation[J]. Journal of machine Learning research, 2003, 3(1):993-1022.

[9] Brin, S. What can you do with a Web in your Pocket?[J]. IEEE Data Eng. Bull., 1998, 21(2):37-47.
[10] Arora, R, and B. Ravindran. Latent dirichlet allocation based multi-document summarization[C]. Proceedings of the second workshop on Analytics for noisy unstructured text data, 2008:91-97.

[11] Vanderwende, L. Beyond SumBasic: Task-focused summarization with sentence simplification and lexical expansion[J]. Information Processing & Management, 2007, 43(6):1606-1618.

[12] Gupta, S, A. Nenkova, and D. Jurafsky. Measuring importance and query relevance in topic-focused multi-document summarization[C]. Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions Association for Computational Linguistics, 2007, 27(1):193-196.

[13] Lin, C Y. Rouge: a package for automatic evaluation of summaries[C]. Proceedings of the Workshop on Text Summarization Branches Out, 2004:25-26.