Exploiting Conceptual Relations of Sentences for Multi-document Summarization

Hai-Tao Zheng(✉), Shu-Qin Gong, Ji-Min Guo, and Wen-Zhen Wu

Tsinghua-Southampton Web Science Laboratory, Graduate School at Shenzhen,
Tsinghua University, Shenzhen, China
{zheng.haitao,guojm14,wuwz12}@sz.tsinghua.edu.cn,
gongshuqin90@gmail.com

Abstract. Multi-document Summarization becomes increasingly important in the age of big data. However, existing summarization systems do not or implicitly consider the conceptual relations of sentences. In this paper, we propose a novel method called Multi-document Summarization based on Explicit Semantics of Sentences (MDSES), which explicitly take conceptual relations of sentences into consideration. It is composed of three components: sentence-concept graph construction, concept clustering and summary generation. We first obtain sentence-concept semantic relation to construct a sentence-concept graph. Then we run graph weighting algorithm to get ranked weighted sentences and concepts. Besides, we obtain concept-concept semantic relation for concepts clustering to eliminate redundancy. Finally, we conduct summary generation to get informative summary. Experimental results on DUC dataset using ROUGE metrics demonstrate the good effectiveness of our methods.

Keywords: Multi-document summarization · Sentence-concept graph · Concept clustering · Summary generation

1 Introduction

Multi-document summarization is to produce a summary from a set of documents which describe the same topic, and a variety of document summarization methods have been developed recently [1,2]. As documents and concepts have the same characteristic that focus on an issue, documents are the reflection of concepts to some extent. However, existing methods do not or implicitly reflect the conceptual relation of sentences, while we explicitly construct the relation between sentences and concepts, which better reflect the relations of sentences in the concept degree. In this paper, we propose a novel method called Multi-document Summarization based on Explicit Semantics of Sentences (MDSES). The contributions of our work are summarized as follows: 1) We explicitly consider the conceptual relations of sentences in the task of multi-document summarization. 2) We propose a novel method called MDSES which utilizes explicit semantics of sentences. 3) We exploit sentence-concept semantic relation and concept-concept semantic relation which is based on Wikipedia textual content and hyperlink structure to eliminate redundancy. 4) Experimental results on the DUC dataset verify the effectiveness of MDSES compared with baselines.

J. Li and Y. Sun (Eds.): WAIM 2015, LNCS 9098, pp. 506–510, 2015.
DOI: 10.1007/978-3-319-21042-1_51

2 Multi-document Summarization Based on Explicit Semantics of Sentences

MDSES is composed of three components: sentence-concept graph generation, concept clustering and summary generation. First, we parse $D = \{d_1, d_2, ..., d_l\}$ to sentence set $S = \{s_1, s_2, ..., s_n\}$. Then we map sentences to Wikipedia concepts based on Wikipedia textual content, and we can get concept set $C = \{c_1, c_2, ..., c_m\}$. After the mapping procedure, we can get sentence-concept relation M_{sc}. We exploit M_{sc} to construct sentence-concept graph $G = \{S, C, E\}$. Then we run graph weighting algorithm on G to get ranked weighted sentence set S' and concept set C'. Since similar sentences map to similar concepts, we need cluster concepts to detect redundancy. First, we compute concept-concept relation M_{cc} based on Wikipedia hyperlink structure. Then we clustering concepts based on M_{cc} to concept clusters CC. At last we conduct summary generation to generate summary. Fig.1 presents the overall framework of MDSES.

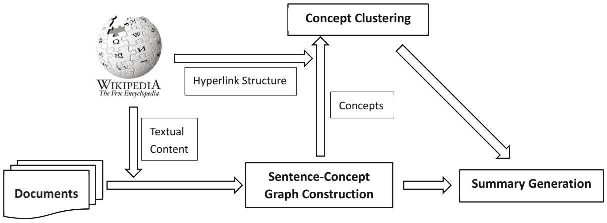


Fig. 1. The framework of MDSES method

Sentence-Concept Graph Construction: We use Explicit Semantic Analysis (ESA) [3] to mapping sentences to Wikipedia concepts. First, we build an inverted index of Wikipedia, which maps each term into a list of concepts in which it appears. We define $s_i = \{t_1, t_2, ..., t_n\}$ be input text, and define $\langle v \rangle = \{v_1, v_2, ..., v_n\}$ be its TFIDF vector, where v_i is the weight of term t_i. let $\langle k \rangle = \{k_1, k_2, ..., k_m\}$ be an inverted index entry for term t_i, where k_j quantifies the strength of association of term t_i with Wikipedia concept c_j, $\{c_j \in c_1, ..., c_m\}$. Then, the weight w_{ij} between s_i and c_j is defined as $\sum_{t_i \in s_i} v_i * k_j$. After the mapping procedure, we can get a sentence-concept relation represented by M_{sc}. Then we use S and C as vertex, M_{sc} as the weighted edge E to get the sentence-concept graph $G = \{S, C, E\}$. Define $weight(s)$ as the weight of s and $weight(c)$ as the weight of c, and we initialize each s in S with the score $\frac{1}{\sqrt{n}}$, then we calculate the weight of c_i and s_j iteratively as follows: $weight^{(k+1)}(c_i) = \sum_{s_j \in S} w_{ji} weight^{(k)}(s_j)$, $weight^{(k+1)}(s_j) = \sum_{c_i \in C} w_{ji} weight^{(k)}(c_i)$. In order to guarantee the convergence of the iterative, the weight of vertex is normalized after each iteration. We iterate the procedure until it reaches convergence, and we can get ranked weighted sentence set S' and ranked weighted concept set C'.

Concept Clustering: In order to get concept-concept relation M_{cc}, we adopt Wikipedia Link Vector based Measure (WLVM) [4]. For each concept, we build a vector space model, which using link counts weighted by the probability of each link occurring. This probability is defined by the total number of links to the target concept over the total number of concepts. Thus if t is the total number of concepts within Wikipedia, then the weighted value w for the link $a \rightarrow b$ (a is the source concept and b is the target concept) is defined as $w(a \rightarrow b) = |a \rightarrow b| * log(\sum_{x=1}^{t} \frac{t}{|x \rightarrow b|})$. We define all n target concepts $\{l_i | i = 1..n\}$ found within the links contained in concepts c_1 and c_2. The vector for each concept c_i is given by $\overrightarrow{c_i} = \{w(c_i \rightarrow l_1), w(c_i \rightarrow l_2), ..., w(c_i \rightarrow l_n)\}$. Our similarity measure for the concepts is then given by the cosine similarity between their vectors. After we compute the similarity between concepts, we can obtain M_{cc}. For concept clustering, we use the conventional Hierarchical Agglomerative Clustering (HAC) algorithm to cluster concepts based on M_{cc} and get concept clusters $CC = \{cc_1, cc_2, ..., cc_k\}$. In HAC algorithm, we finished the cluster merging when the similarity between concept clusters falls below a given threshold (0.3 is an empiric value in our study).

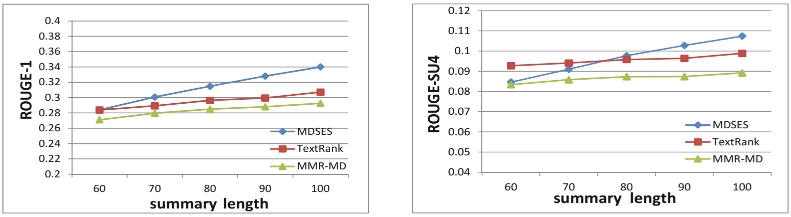
Summary Generation: First, we rank CC based on the ranked weighted C'. For each cluster cc, we compute its weight $weight(cc) = \sum_{c_i \in cc} weight(c_i)$. Then we rank concept clusters in descending order of weight to get ranked concept clusters CC'. For each cluster, we select one concept c with the highest weight to represent the cluster based on the ranked C', and we can get representative ranked concept set $RC = \{c_1, c_2, ..., c_k\}$. At last, we select the first concept c from RC, and get sentence set S_c linked to c based on G, then we select the sentence with the highest weight in S_c based on S' to generate summary. Iterate the procedure until the length of summary is reached.

3 Experiment

In our experiments, we use DUC[1] 2004 dataset and ROUGE [5] as evaluation metric. For baselines, we choose MMR-MD [6] and TextRank [1] for they are two classical methods. With the ubiquity of mobile internet and people's reading habit on mobile devices, different length of summaries are also important. We investigate the performances of MDSES under different summary length, which ranging from 60 words to 100 words. For evaluation, we use ROUGE-1 and ROUGE-SU4, and Fig.2 are comparison at different summary length.

From Fig.2 we have three obvious: (1)All the curves are incremental curves and the reason is also obvious. It is because that with the length of summary increases, the summary will contain more important information. (3)The gradient of MDSES is higher than TextRank and MMR-MD, which means that our

[1] http://duc.nist.gov/



(a) ROUGE-1. (b) ROUGE-SU4.

(a) ROUGE-1 (b) ROUGE-SU4

Fig. 2. Comparison at different summary lengths

MDSES tends to have more advantages when the length of summary increases. With the summary length increases, TextRank and MMR-MD select redundancy sentences to summary while MDSES selects sentences in concept degree which leading to more coverage of summary. (3)TextRank outperforms MDSES in Fig. 2(b) when the summary length is less than 80 words. The reason may be that the summary length is too small, the importance of conceptual relations of sentences is restricted. But while the summary length is appropriate or big, the good effect of exploiting conceptual relations of sentences for multi-document summarization appears.

4 Conclusion

In this paper we propose a novel method MDSES, it contains three components: sentence-concept graph construction, concept clustering and summary generation. Unlike with other methods, we explicitly construct the relation between sentences and concepts, which better reflect the relations of sentences in the concept degree. Experimental results verify the effectiveness of MDSES.

Acknowledgments. This research is supported by the 863 project of China (2013AA013300), National Natural Science Foundation of China (Grant No. 61375054) and Tsinghua University Initiative Scientific Research Program (Grant No.20131089256).

References

1. Mihalcea, R., Tarau, P.: TextRank: Bringing order into texts. Association for Computational Linguistics (2004)
2. Gong, S., Qu, Y., Tian, S.: Summarization using wikipedia. In: TAC 2010 Proceedings (2009)
3. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using wikipedia-based explicit semantic analysis. In: IJCAI, vol. 7, pp. 1606–1611 (2007)

4. Milne, D.: Computing semantic relatedness using wikipedia link structure. In: Proceedings of the New Zealand Computer Science Research Student Conference (2007)
5. Lin, C.Y.: Rouge: a package for automatic evaluation of summaries. In: Text Summarization Branches Out: Proceedings of the ACL 2004 Workshop, pp. 74–81 (2004)
6. Goldstein, J., Mittal, V., Carbonell, J., Kantrowitz, M.: Multi-document summarization by sentence extraction. In: Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic Summarization, vol. 4, pp. 40–48 (2000)