Postprint: Core Topic Sentence Identification in Academic Papers Using an Improved TextRank Algorithm Based on WMD Semantic Similarity

Authors: Wang Zixuan, Le Xiaoqiu, He Yuanbiao

Date: 2017-11-08T00:00:00+00:00

Abstract

【Objective】 To automatically identify key sentences describing research topics in scientific papers. 【Method】 Organizing sentence sets by paper subsections, computing semantic similarity between sentences through WMD distance using trained domain-specific word embeddings, optimizing the iterative process of the TextRank algorithm, adjusting the obtained weights using external features, and selecting key topic sentences in descending order of sentence weights. 【Results】 Using scientific papers in the climate change domain as experimental data and manually annotated results as a benchmark, comparative experiments were conducted between the proposed algorithm and the traditional TextRank algorithm. Preliminary results show that the recognition performance (F-score) of this method improves by approximately 5% over the traditional TextRank algorithm. 【Limitations】 Sentence feature extraction needs improvement, and word embedding training as well as relevant parameters in the method require further optimization. 【Conclusion】 The improved TextRank algorithm based on domain-specific word embeddings and incorporating WMD semantic similarity can effectively identify central sentences within subsections of scientific papers, and with weight adjustment assisted by external features, can effectively identify the core topic sentences of a paper.

Full Text

Recognizing Core Topic Sentences with Improved TextRank Algorithm Based on WMD Semantic Similarity

Wang Zixuan[1],[2], **Le Xiaoqiu**[1], **He Yuanbiao**[1]
[1](National Science Library, Chinese Academy of Sciences, Beijing 100190, China)
[2](University of Chinese Academy of Sciences, Beijing 100049, China)

Abstract

[**Objective**] This paper aims to automatically identify key sentences that describe research topics in scientific papers. [**Methods**] We organized sentence sets by paper sections, calculated WMD distances between sentences using trained domain word embeddings to obtain semantic similarity measures, optimized the TextRank algorithm's iterative process, and adjusted the resulting weights using external features to select key topic sentences in descending order of sentence weight. [**Results**] Using scientific papers on climate change as experimental data and comparing our algorithm with traditional TextRank based on manually annotated results, preliminary experiments show that the recognition effectiveness (F-value) of our method improves by approximately 5% over traditional TextRank. [**Limitations**] Sentence feature extraction needs improvement, and both word embedding training and related parameters in the method require further optimization. [**Conclusions**] The improved TextRank algorithm based on domain word embeddings and WMD semantic similarity can effectively identify central sentences within scientific paper sections, and can recognize core topic sentences of an entire paper after weight adjustment with external features.

Keywords: WMD; TextRank; Semantic Similarity; Topic Sentence Recognition; External Features
Classification Number: TP393

---

1. Introduction

Authors of scientific papers typically focus on a primary research problem, which can be represented as a research topic in literature analysis. Topic sentences are those used to argue and support the research topic, distributed throughout the main paragraphs of the paper. As a fundamental text analysis technique, topic sentence recognition plays a crucial role in natural language processing applications such as information retrieval, automatic summarization, and knowledge discovery. Identifying core topic sentences in domain-specific scientific papers involves distinguishing and extracting key statements that describe and reveal the research topic from the full text. This is a critical technical step in distilling scientific paper content, helping researchers quickly discover relatively valuable content and improving research efficiency.

The general process of text topic sentence identification involves: (1) identifying candidate topic sentences in the text, and (2) reasonably evaluating the importance of these candidates in expressing core content and themes, then selecting appropriate sentences as topic sentences [**?**]. Methods for evaluating sentence importance primarily involve measuring both intrinsic sentence features (position, topic words, length, etc.) and inter-sentence relationships. The former uses statistical features to build models for weight scoring or supervised learning, while the latter transforms sentences and their relationships into graph

models for recognition, exemplified by TextRank [**?**].

Traditional TextRank uses feature word vectors to represent sentences and calculates inter-sentence similarity using distance metrics such as Euclidean distance or cosine similarity. However, this approach suffers from the curse of dimensionality and synonym/hyponym problems in sentence representation. To address these issues, this paper employs WMD (Word Mover's Distance) [**?**] based on word embedding semantic similarity to represent inter-sentence distances, improves the TextRank algorithm, and optimizes the results using paper content structure to update weights and rankings, ultimately obtaining core topic sentences of scientific papers.

---

2. Related Research on Topic Sentence Recognition

As a foundational task for many natural language processing applications, scholars have proposed various methods for topic sentence recognition. Due to different developmental periods and technical approaches, these mainly fall into three categories:

(1) Statistical Feature-Based Methods. These methods transform source texts into linear sequences of sentences and sentences into linear sequences of words, assigning weights to words and sentences based on certain feature indicators, and finally selecting sentences with higher comprehensive weights as output [**?**]. Luhn [**?**] identified word frequency, Baxendale [**?**] noted sentence position, and Liu et al. [**?**] summarized information such as titles, positions, and syntactic structures as indicators for measuring sentence importance. Edmundson [**?**] selected several variables to construct a simple multivariate linear function: $Weight_x = C + K + T + L$, where C, K, T, and L represent four feature variables with others as adjustment parameters, using multiple features to describe sentence weights. Practice shows this representation is not ideal, as the linear addition process lacks theoretical foundation. Statistical feature methods are simple and fast but heavily influenced by feature selection and weighting approaches, resulting in unstable performance.

(2) Machine Learning Binary Classification Methods. These approaches transform topic sentence recognition into a binary classification problem at the sentence level, involving four steps: feature selection, algorithm selection, model training, and topic sentence discrimination. Algorithms applicable to topic sentence recognition include Naive Bayes, Conditional Random Fields, Support Vector Machines, and other models. Kupiec et al. [**?**] first used an NB classifier for topic sentence identification with features including sentence length, fixed phrases, paragraphs, topic words, and capitalized words. Conroy et al. [**?**] applied HMM to topic sentence recognition, using observed sequences to find the most likely hidden state sequence, constructing observation state transition probability matrices from three document features (sentence position, word frequency, and word probability) to build prediction models. While effective, ma-

chine learning classification methods require substantial training data and rely on feature independence assumptions, limiting their applicability and operability.

(3) Graph-Based Ranking Methods. These decompose text into sentence units, with each unit corresponding to a vertex in a graph structure and similarity relationships between units serving as edges. Graph ranking algorithms then calculate vertex scores, selecting higher-scoring sentences as topic sentences. Different graph ranking methods vary primarily in edge weight calculation and algorithm selection. Common edge weight calculations include word co-occurrence and sentence similarity, while ranking algorithms include matrix weight addition and PageRank. Mihalcea et al. [**?**] first proposed the TextRank algorithm for topic sentence recognition and optimized it in subsequent work [**?**]. Yu et al. [**?**] filtered non-important words, merged synonyms, used vector space models to represent sentences, calculated cosine similarity as edge weights, and added manual features to optimize TextRank results. Geng et al. [**?**] and He et al. [**?**] utilized topic information from word co-occurrence and connection features between different topics to identify topic sentences. Graph ranking methods achieve results without external knowledge or training samples, but result quality is affected by edge weight calculation methods and remains unstable.

For structured texts like scientific papers where implicit semantic connections exist between sentences, this paper improves sentence similarity calculation methods based on graph ranking algorithms for topic sentence recognition and optimizes results by considering text structural features to enhance recognition performance.

Scientific papers represent summaries of research processes. Saïd et al. [**?**] noted that authors' writing approaches influence article content structure, reflected in relationships between logical elements (titles, paragraphs, sections, etc.). Leveraging both internal connections among paragraph sentences and the overall external structure of papers plays an important role in identifying core components within text paragraphs.

Our proposed method for recognizing core topic sentences in papers includes four steps: word vector representation and training, inter-sentence similarity calculation, TextRank algorithm iteration, and result optimization using external structural features, as shown in [Figure 1: see original paper].

First, we train domain word vectors on full-text scientific papers using the Word2Vec model. Second, we segment full papers into sentences, remove meaningless short sentences, and use the trained word vectors to represent WMD distances between sentences, converting them to similarity measures. Third, we construct undirected weighted graphs for each text section, using inter-sentence similarity as edge weights, and iterate with the TextRank algorithm to obtain sentence weight values. Finally, we adjust and rank the weights using feature information such as sentence position and outline structure, ultimately identifying

core topic sentences proportionally.

3.1 Improved TextRank Algorithm Based on WMD Semantic Similarity

(1) Word Vector Representation and Training. Words are the most basic units carrying semantic information. Traditional one-hot representation isolates each word using 0s and 1s, resulting in vectors without semantic information and suffering from the curse of dimensionality. Harris's Distributional Hypothesis [**?**] suggests that word semantics are determined by their context. Bengio et al. [**?**] proposed the Neural Network Language Model (NNLM), which models relationships between target words and complex contexts, obtaining low-dimensional byproducts—Word Embeddings—while learning language models. Traditional NNLM models have high computational complexity and low efficiency for large datasets. Mikolov et al. [**?**] removed hidden layers and proposed CBOW (Continuous Bag-of-Words) and Skip-gram models. CBOW predicts a word from its context, while Skip-gram does the opposite, using the current word's vector as input to output surrounding words' vectors. Training involves two optimization methods: Hierarchical Softmax, which converts the original Softmax to a hierarchical version with Huffman trees, reducing prediction time by log n times using Huffman tree properties; and Negative Sampling, which uses simpler random weighted negative sampling to significantly improve performance and speed. These models and methods can be arbitrarily combined, with complete implementations available in Google's Word2Vec [**?**] open-source toolkit released in 2013.

(2) Sentence Similarity Calculation. Traditional sentence similarity calculation methods include: (a) feature word-based methods using TF-IDF, chi-square values, mutual information to select feature words and build vectors for calculation [**?**]; (b) syntactic parsing-based methods that analyze sentence syntax to calculate structural and content similarity, currently focusing on simple syntactic structure matching [**?**]; and (c) semantic analysis-based methods that use semantic dictionaries or ontologies for word sense disambiguation and consider semantic similarity during calculation [**?**]. The first two methods suffer from dimensionality disaster and synonym problems, while the third relies on external knowledge whose quality and coverage directly affect results, lacking scalability.

Word embeddings effectively solve these problems with simple training processes and capture potential semantic relationships between words. Mikolov et al. [**?**] discovered vector arithmetic relationships such as $c_{king} - c_{woman} + c_{queen} \approx c_{man}$. Directly summing and normalizing word vectors in a sentence yields sentence vectors with effective cosine similarity calculations. Kusner et al. [**?**] proposed Word Mover's Distance (WMD), where word-to-word similarity is represented by Euclidean distance and sentence-to-sentence similarity is transformed into

a transportation optimization problem, viewing inter-sentence similarity as a transformation between two probability distributions with distance represented by transformation cost, achieving good results in KNN text classification. The paper also proved that averaging word vectors and calculating Euclidean distance is a lower bound of WMD.

For WMD distance calculation between sentences s and s', we first transform both sentences into word bags, remove stop words (totaling n words), normalize remaining word frequencies to build word frequency vectors denoted as $d, d' \in \mathbb{R}^n$, calculate Euclidean distances between every word pair as transportation costs to build transfer matrix $T \in \mathbb{R}^{n \times n}$, and formulate cost minimization as shown in Equation (1):

$$\begin{aligned} &\min_{T \geq 0} \sum_{i,j=1}^{n} T_{ij} \cdot c(i,j) \\ &\text{s.t.} \sum_{j=1}^{n} T_{ij} = d_i \quad \forall i \in \{1, \ldots, n\} \\ &\sum_{i=1}^{n} T_{ij} = d'_j \quad \forall j \in \{1, \ldots, n\} \end{aligned}$$

We solve this using the EMD algorithm [**?**], with the transformation process shown in [Figure 2: see original paper]. Sentence similarity is calculated similarly to converting Euclidean distance to similarity, yielding the final inter-sentence similarity measure shown in Equation (2):

$$sim(s, s') = \frac{1}{1 + wmd(s, s')}$$

(3) TextRank Algorithm Iteration. A document typically consists of multiple paragraphs, with consecutive paragraphs forming semantically cohesive sections corresponding to sub-topics organized under higher-level topics in the outline. Sentences from content-similar paragraphs within the same section form independent sentence groups. Building network graphs at the sentence group level for topic sentence recognition effectively identifies sentences representing entire section content, ensuring recognition quality.

The TextRank method borrows from PageRank, viewing inter-sentence similarity as a support or recommendation relationship. Sentences serve as nodes, with similarity measures as edges to build graph models. Through iterative calculations, we optimize sentence weights and select higher-weight sentences as topic sentences.

Assuming a text section comprises n sentences $V = \{V_1, V_2, \ldots, V_n\}$, we build a TextRank graph $G = (V, E)$ with sentences as nodes and similarity relationships

as edges. From previous calculations, we obtain the $n \times n$ sentence similarity matrix S shown in Equation (3):

$$S = \begin{bmatrix} sim(V_1, V_1) & sim(V_1, V_2) & \cdots & sim(V_1, V_n) \\ sim(V_2, V_1) & sim(V_2, V_2) & \cdots & sim(V_2, V_n) \\ \vdots & \vdots & \ddots & \vdots \\ sim(V_n, V_1) & sim(V_n, V_2) & \cdots & sim(V_n, V_n) \end{bmatrix}$$

We iteratively calculate each node's weight using G and S as shown in Equation (4):

$$WS(V_i) = (1-d) + d \times \sum_{V_j \in In(V_i)} \frac{S_{ji}}{\sum_{V_k \in Out(V_j)} S_{jk}} WS(V_j)$$

where $WS(V_i)$ is node V_i's weight, $In(V_i)$ represents nodes pointing to V_i, $Out(V_i)$ represents nodes V_i points to, and d is the damping coefficient (typically 0.85). Initial node weights are usually set to $1/n$. Iteration stops when weight changes between two iterations become very small and approach zero, yielding final sentence weights for ranking and proportional selection of top-weighted sentences as topic sentences.

3.2 External Feature Selection and Weight Calculation

Sentence weights from iterative calculation converge to stable values determined by other sentences, independent of initial values, making pre-iteration weight adjustments meaningless. After TextRank completion, we adjust the resulting sentence weight sequence using external features:

(1) Sentence Position. Baxendale [**?**] statistically found that paragraph topic sentences appear as first sentences with 85% probability and as last sentences with 7% probability. Similarly, for paragraphs within sections, opening and closing paragraphs are more likely to guide or summarize section content and reveal topics. We therefore weight sentences based on both paragraph position within sections and sentence position within paragraphs, with greater weight enhancement for sentences in first/last paragraphs and first/last sentences. Both weighting methods use the same function shown in Equation (5):

$$f(x) = p \cdot e^{-x^2}$$

where x represents the percentage position of a sentence within its paragraph or a paragraph within its section (0 to 1 from beginning to end), with weight enhancement ratios denoted as p_1 and p_2. Final weight adjustment becomes: $WS'(V_i) = WS(V_i) \times (1 + f_{para}(x) + f_{sent}(x))$.

(2) Core Terminology. While TextRank extracts sentences relatively expressing section content from within sentence groups, they may not necessarily relate to the paper's overall theme. We therefore optimize using external features like article titles, keywords, and outlines relative to sentence groups. Using He et al.'s [**?**] outline term extraction method, we identify terms from titles and outline hierarchical structures, merging them with keywords to obtain a core terminology set. Sentences better reflecting core terminology are more likely to be core topic sentences. Current solutions only consider inclusion relationships, where p_3 is the weighting factor (set to 0.1) and n is the count of included core terms: $WS'(V_i) = WS(V_i) \times (1 + p_3 \cdot n)$.

(3) Sentence Category. Literature [**?**] notes that outlines contain not only specific terms but also many broadly meaningful paper terms like "method," "conclusion," etc., representing facet descriptions for research points that can serve as topic description framework indicators. For full paper content, sentences with certain categories hold greater value in elaborating topic-related facet descriptions. Recognizing such categorized sentences aligns with structured abstract generation. We therefore weight sentences with certain categories, particularly for outlines and text sections containing paper terminology, where corresponding category sentences represent concentrated content expression and deserve greater weighting: $WS'(V_i) = WS(V_i) \times (1 + p_4 \cdot n \cdot b)$, where p_4 is the weighting factor (set to 0.1), n is sentence category count, and b indicates whether the sentence contains paper terminology categories (0 or 1). Sentence classification can be addressed using traditional models like Naive Bayes, CRF, or deep neural networks like LSTM and GRU [**?**], which we won't elaborate here.

4.1 Experimental Process

To validate our topic sentence identification method, we conducted experiments using climate change domain data. The dataset comprised 31,430 full-text papers from 10 journals including *Atmospheric Research* downloaded from Elsevier.

We extracted titles, abstracts, outlines, and full texts to build word embedding training corpora. Based on data volume and NLP tasks, we selected appropriate training models, optimization algorithms, and hyperparameters. We chose the Skip-gram model with Hierarchical Softmax, which better represents uncommon domain terms. Other hyperparameters included a context window of 5 and word vector dimension of 100. After 5 hours of training, we obtained a word vector file of approximately 400MB.

We segmented full papers into sentences, extracted textual features including position, section, and outline information, and built WMD-based semantic similarity between sentences using the word vector file. We then identified topic sentences using the improved TextRank algorithm. The test corpus consisted of 9 full-text papers with topic sentences annotated by domain experts.

4.2 Results Analysis

In addition to our method, we implemented traditional TextRank, WMD matrix addition, and WMD+TextRank methods using the same training corpus and test documents for comparative analysis, with results shown in .

The results show that although our method performed slightly better than others within the same text sections, all four methods yielded unsatisfactory results. Analyzing the experimental data and results revealed the following reasons: (1) The test set focused on El Niño phenomenon data with low correlation to word vector training corpora, resulting in poor vector representations for some terms that affected sentence similarity calculation; (2) The test set was annotated by a single domain expert, potentially introducing accuracy bias; (3) Using sections as basic recognition units with fixed proportions assumed uniform distribution of core topic sentences, whereas actual distribution varies across paper sections—introduction, experimental results, and conclusion sections contain higher densities of core topic sentences, while literature review and methodology sections contain lower densities.

We therefore conducted additional experiments using computer science data as word vector training corpora. The experimental process remained identical, but we used multi-person collaborative annotation to select mutually agreed sentences and doubled the recognition proportion for paper beginning and end sections. Results are shown in .

Method	Precision	Recall	F1-Score
TextRank	25.05%	38.59%	30.37%
WMD+TextRank	27.66%	42.59%	33.54%
Our Method (WMD+TextRank+External Features)	29.06%	44.75%	35.24%

These adjustments improved F1-score by nearly 10% compared to the previous experiment, with our method achieving nearly 5% higher F1-score than traditional TextRank, yielding relatively better results.

Summarizing the experimental process and results, we draw the following conclusions: (1) Traditional TextRank shows stable performance but tends to identify longer sentences due to similarity calculation limitations; (2) Word vector quality affects sentence similarity calculation and thus our method's results, while traditional TextRank's co-occurrence-based calculation is less affected by poor word vectors; (3) Although our method's recognition effectiveness still has room for improvement, analysis shows that unselected sentences are generally also valuable, with overall quality better than the other three methods; (4) Among unselected sentences, some contain special cue words (e.g., "Hence," "In this paper," "It shows that") or numerical data—前者 indicate important

summary statements authors declare, while 后者 represent the paper's most convincing evidence, both playing crucial roles in demonstrating core themes. Our graph model approach weakens this information, causing these sentences to be missed. Further research on extracting these features could optimize the method and improve recognition effectiveness.

In summary, our proposed topic sentence recognition method requires no external knowledge structures. It uses full-text word embeddings to improve similarity calculation, refines the TextRank iteration process, adjusts result weights based on external features, and leverages both internal sentence connections and external paper structure information. The recognition effectiveness reaches average human evaluation levels, outperforming previous work [**?**], though significant improvements remain possible.

Conclusion

This paper analyzed mainstream topic sentence extraction methods and, addressing characteristics of scientific papers, proposed an improved TextRank algorithm incorporating WMD semantic similarity based on domain word embeddings. Experimental results demonstrate that our method can effectively identify central sentences within paper sections and recognize core topic sentences of entire papers after external feature weight adjustment. However, limitations remain in sentence feature extraction and word vector training processes, and method parameters require further optimization. Future work will continue optimizing parameters, extracting more effective sentence features, and leveraging word embeddings to discover latent relationships between core vocabulary and sentences to improve core topic sentence recognition accuracy.

References

[1] Sunayama W, Yachida M. Panoramic View System for Extracting Key Sentences Based on Viewpoints and Application to a Search Engine[J]. Journal of Network and Computer Applications, 2005, 28(2): 115-127.
[2] Mihalcea R, Tarau P. TextRank: Bringing Order into Texts[OL]. Unt Scholarly Works, 2004. https://digital.library.unt.edu/ark:/67531/metadc30962/m2/1/high_{res}d/Mihalcea-2004-TextRank-Bringing_{{{Order}}*{{into}}*}{Texts}.pdf.
[3] Kusner M J, Sun Y, Kolkin N I, et al. From Word Embeddings to Document Distances[C]//Proceedings of the 32nd International Conference on Machine Learning. 2015: 957-966.
[4] Batcha N K, Aziz N A. An Algebraic Approach for Sentence Based Feature Extraction Applied for Automatic Text Summarization[J]. Journal of Computational & Theoretical Nanoscience, 2014, 20(1): 139-143.
[5] Luhn H P. The Automatic Creation of Literature Abstracts[J]. IBM Journal of Research and Development, 1958, 2(2): 159-165.

[6] Baxendale P B. Machine-Made Index for Technical Literature——An Experiment[J]. IBM Journal of Research and Development, 1958, 2(4): 354-361.
[7] Liu Ting, Wang Kaizhu. Four Kinds of Main Methods of Automatic Abstracting[J]. Journal of the China Society for Scientific and Technical Information, 1999, 18(1): 10-19.
[8] Edmundson H P. New Methods in Automatic Extracting[J]. Journal of the ACM, 1969, 16(2): 264-285.
[9] Kupiec J, Pedersen J, Chen F. A Trainable Document Summarizer[C]//Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 1995: 68-73.
[10] Conroy J M, O'leary D P. Text Summarization via Hidden Markov Models[C]//Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2001: 406-407.
[11] Mihalcea R. Graph-based Ranking Algorithms for Sentence Extraction, Applied to Text Summarization[C]//Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions. Association for Computational Linguistics, 2004: 20-24.
[12] Yu Shanshan, Su Jindian, Li Pengfei. Improved TextRank-based Method for Automatic Summarization[J]. Computer Science, 2016, 43(6): 240-247.
[13] Geng Huantong, Cai Qingsheng, Zhao Peng, et al. A Kind of Automatic Text Keyphrase Extraction Method Based on Word Co-occurrence[J]. Journal of the China Society for Scientific and Technical Information, 2005, 24(6): 651-656.
[14] He Wei, Wang Yu. Extracting Topic Sentences from Web Text Based on Sentence Relationship Map[J]. New Technology of Library and Information Service, 2009(3): 57-61.
[15] Saïd T, Evrard F. Intentional Structures of Documents[C]//Proceedings of the 12th ACM Conference on Hypertext and Hypermedia. ACM, 2001: 39-40.
[16] Harris Z S. Distributional Structure[A]//Papers on Syntax[M]. Springer Netherlands, 1954.
[17] Bengio Y, Schwenk H, Senécal J S, et al. A Neural Probabilistic Language Model[J]. Journal of Machine Learning Research, 2003, 3(6): 1137-1155.
[18] Mikolov T, Sutskever I, Chen K, et al. Distributed Representations of Words and Phrases and Their Compositionality[J]. Advances in Neural Information Processing Systems, 2013, 26: 3111-3119.
[19] Word2Vec[EB/OL]. [2016-12-26]. https://code.google.com/p/word2vec/.
[20] Guo Qinglin, Li Yanmei, Tang Qi. Similarity Computing of Documents Based on VSM[J]. Application Research of Computers, 2008, 25(11): 3256-3258.
[21] Wang R, Neumann G. Recognizing Textual Entailment Using Sentence Similarity Based on Dependency Skeletons[C]//Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing. Association for Computational Linguistics, 2007: 36-41.
[22] Wang D, Li T, Zhu S, et al. Multi-document Summarization via Sentence-level Semantic Analysis and Symmetric Matrix Factorization[C]//Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2008.

[23] Mikolov T, Chen K, Corrado G, et al. Efficient Estimation of Word Representations in Vector Space[OL]. arXiv Preprint. arXiv: 1301.3781, 2013.
[24] Ling H, Okada K. An Efficient Earth Mover's Distance Algorithm for Robust Histogram Comparison[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(5): 840-853.
[25] He Yuanbiao, Le Xiaoqiu, Zhang Fan. Research on Keyphrase Extraction from Scholarly Article Outline[J]. New Technology of Library and Information Service, 2014(3): 73-79.
[26] He Yuanbiao. Phrase Hierarchical Relationship Mining Based on Scholarly Article Outline[D]. Beijing: University of Chinese Academy of Sciences, 2014.
[27] Chung J, Gulcehre C, Cho K H, et al. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling[OL]. arXiv Preprint. arXiv: 1412.3555, 2014.

Author Contributions

Wang Zixuan: Designed and implemented the technical solution and roadmap, collected and cleaned data, conducted experimental analysis and validation, drafted and wrote the paper, and revised the final version.
Le Xiaoqiu: Proposed the research direction and main ideas, optimized research design and technical roadmap, and revised the paper.
He Yuanbiao: Implemented some modules and participated in paper revision.

Conflict of Interest Statement

All authors declare no conflict of interest.

Supporting Data

The supporting data is self-archived by the authors and available upon request at E-mail: lexq@mail.las.ac.cn.

[1] Wang Zixuan, Le Xiaoqiu, He Yuanbiao. rec_{result}.xlsx. Comparative dataset of topic sentence recognition results in climate change and computer science domains.

Received: 2017-01-19
Revised: 2017-03-13

Note: Figure translations are in progress. See original paper for figures.