Research on Automatic Indexing Methods for Scientific Literature Based on Multi-Feature Fusion

Authors: Jiachen Ju, Song Peiyan, Ju Jiachen

Date: 2022-09-02T00:00:00+00:00

Abstract

Purpose/Significance In the current era where users urgently need to efficiently acquire valuable information from extremely complex information environments, this paper proposes an automatic indexing method based on multi-feature fusion to enhance the accuracy of text indexing. Method/Process First, both the text body and abstract are simultaneously employed as indexing sources. Next, the Keybert method and TF-IDF method are respectively applied to process the abstract and body, while combining the term frequency features from statistical learning approaches and the semantic features from machine learning approaches to obtain two sets of candidate indexing terms. Finally, a fusion process is conducted through semantic similarity computation, integrating the advantages of both methods to achieve an overall balance between accuracy and comprehensiveness in the indexing results. Results/Conclusion Experimental results indicate that text automatic indexing based on multi-feature fusion is feasible and achieves satisfactory indexing performance.

Full Text

Research on Automatic Indexing of Scientific and Technological Documents Based on Multi-Feature Fusion

Ju Jiachen; Song Peiyan
Tianjin Normal University, Tianjin 300387

Abstract

[**Purpose/Significance**] Currently, users urgently need to efficiently acquire valuable information from extremely complex information environments. In

this context, this paper proposes an automatic indexing method based on multi-feature fusion to improve the accuracy of text indexing. [**Method/Process**] First, both the full text and abstract are simultaneously used as indexing sources. The KeyBERT method and TF-IDF method are then respectively applied to process the abstract and full text, combining the word frequency features of statistical learning methods with the semantic features of machine learning methods to obtain two sets of candidate indexing terms. Finally, fusion processing through semantic similarity calculation integrates the advantages of both methods to achieve an overall balance of accuracy and comprehensiveness in indexing results. [**Result/Conclusion**] Experiments demonstrate that text automatic indexing based on multi-feature fusion is feasible and produces satisfactory indexing results.

Keywords: automatic indexing; multi-feature fusion; candidate word extraction
Classification Number: G353

With the advent of the big data era, users urgently need to efficiently obtain valuable information from extremely complex information environments, addressing the contradiction between the unlimited growth of information resources and inefficient information retrieval. Keywords serve as an important means for people to quickly understand document content and grasp main topics, widely applied in news and scientific paper domains to facilitate efficient document management and retrieval [1]. Text automatic indexing refers to the process of using computer systems to mimic human indexing activities, automatically annotating keywords from factual information or literature (title, abstract, full text) intended for storage and retrieval. Compared with traditional manual indexing, automatic indexing offers advantages in processing capability, efficiency, low cost, and high consistency and stability, better meeting users' retrieval needs in the information society [2]. With the explosive growth of information in the internet age, text automatic indexing has become a crucial means for users to access core content. Current research on text automatic indexing continues to explore improvements in both accuracy and comprehensiveness.

Text automatic indexing can be divided into automatic term extraction indexing and automatic term assignment indexing [3]. Based on theoretical foundations, automatic indexing can be categorized into statistical analysis methods, linguistic analysis methods, artificial intelligence methods, and hybrid methods [3]. This study employs a term extraction indexing method that combines statistical analysis with deep learning approaches. While statistical learning, linguistic analysis, and artificial intelligence methods each have their own strengths, they also have inherent limitations. Consequently, using hybrid methods for text automatic indexing has become a trend. In summary, considering the limitations of indexing sources and the trade-offs between different indexing methods, this research proposes using both full text and abstract simultaneously as indexing sources, combining word frequency features from statistical learning methods with semantic features from machine learning methods to obtain candidate in-

dexing terms, then performing fusion processing to integrate the advantages of both methods and achieve an overall balance of accuracy and comprehensiveness in indexing results.

2 Related Research

Research on text automatic indexing first flourished abroad, where Luhn [5] pioneered statistical indexing methods based on word frequency, also known as word frequency statistical indexing. China' s research on automatic document indexing began in the early 1980s, starting relatively late and still lagging behind foreign research to some extent [6]. Entering the 21st century, particularly in recent years, related research has gradually expanded, and automatic indexing technology has essentially reached a practical level. Wang Xiaolin [7] incorporated positional features into the TF-IDF algorithm, forming a new keyword extraction method based on word frequency statistics. Jiang Yi, Huang Yong [8] and others fused lexical functional features on the basis of traditional word frequency and positional features, conducting keyword extraction experiments on academic literature in the computer science field using both classification and ranking approaches. To address the drawbacks of inefficient semantic utilization and duplicate semantic extraction, scholars proposed in 2018 a sequence-to-sequence framework with attention, copying, and coverage mechanisms [9]. Li Qianju [10] et al. used string pattern matching methods to automatically extract keywords based on thesauri and keyword vocabularies, effectively optimizing manual indexing results in terms of increment, combination, and ranking. In domestic research on term assignment indexing, Wang Xing and Liu Wei [11] proposed an automatic indexing method for academic literature based on citation relationships between documents using an improved genetic algorithm. Zhang Chengzhi [12] comprehensively examined the impact of eye movement data from general corpora on keyword extraction performance for Weibo posts, from two perspectives: eye movement feature selection and combination of eye movement features with text features, while also proposing an eye movement data expansion scheme to address the significant scale difference between eye movement datasets and test datasets. In summary, China' s research on automatic indexing is gradually deepening, with particularly rapid innovation in indexing methods. Moreover, hybrid methods as an approach to automatic indexing are gaining increasing attention, making the multi-feature fusion approach—a hybrid indexing method—worth exploring in depth for text automatic indexing.

3 Automatic Indexing Model Based on Multi-Feature Fusion

Following the basic approach of introducing multiple features for keyword extraction, the indexing model incorporates as many feature relationships as possible during keyword extraction to improve model accuracy. Text input is divided into short texts such as abstracts and long texts such as full bodies. KeyBERT

and TF-IDF algorithms complement each other' s strengths, with KeyBERT processing short abstract texts and TF-IDF processing long full texts. Indexing term fusion is then achieved through semantic similarity calculation. Therefore, the text automatic indexing model based on multi-feature fusion, given dual text inputs, outputs indexing terms that represent the core content of the text after fusion processing unit computation. The multi-feature fusion text automatic indexing model proposed in this paper is as follows:

[Figure 1: see original paper] Multi-feature fusion method for automatic topic indexing model

As shown in the figure above, the automatic indexing model primarily consists of four components: input, candidate term extraction, candidate term fusion processing, and output. Based on information importance and text length, texts are categorized as short or long. For instance, since an abstract summarizes the full text, keywords are more likely to be extracted from it. Despite its small size, the abstract has high information importance and is therefore classified as short text, while the full text content is classified as long text. The candidate term extraction component performs initial screening of candidate terms from input texts, generating corresponding candidate term sets. KeyBERT algorithm includes preprocessing steps such as word segmentation and stop-word removal in its early stages, so these are not separately listed in the model. The candidate term fusion processing component is responsible for fusing and filtering candidate terms from input texts, with the core of the candidate term fusion processing unit employing the cosine similarity principle. Its structural composition is as follows:

[Figure 2: see original paper] Candidate term fusion processing unit

Fusion processing is performed on candidate term sets generated by different algorithms and different texts. Some vocabulary terms between candidate sets themselves exhibit strong cohesion, with similar meanings. Generally, words with higher similarity are more capable of serving as topic keywords. Extracting and fusing words with high similarity constitutes the core work of the candidate term fusion processing module. As shown in the figure, the solid-line box represents the fusion processing unit structure, while the dashed-line box represents control conditions. Through iterative mutual fusion between word sets and under a determined similarity fusion threshold, words exceeding the threshold are filtered out as candidate terms. Then, based on the final fusion weight sequence condition, top-ranked candidate terms are determined as the final keyword set. Changing the similarity fusion threshold and fusion weight sequence will produce fine-tuning effects on model computation results; optimal parameter settings can be obtained through subsequent experiments. The multi-feature fusion method for automatic topic indexing can approach texts from different perspectives, considering both the importance of short abstract texts and not ignoring key information reflected in long full texts. It combines the efficiency of TF-IDF while fusing semantic features, compensating for the deficiency of insufficient contextual connections.

4 Keyword Extraction Algorithms

4.1 TF-IDF Keyword Extraction

The TFIDF (Term Frequency & Inverse Document Frequency) algorithm, proposed by Stilton [5], is based on the principle that if a word appears more frequently in a specific document, its TF value increases, indicating stronger ability to express that document' s content and thus should be assigned higher weight. Conversely, if a word appears in fewer documents within a collection, its calculated IDF value increases, indicating stronger ability to distinguish document content and should also be assigned higher weight. The TF-IDF calculation formula is as follows:

$$TFIDF(t,d) = TF(t,d) \times IDF(t)$$

In the above formula, t represents a term, d represents a document, TF(t, d) represents the frequency of term t in document d, and DF(t) represents the number of documents containing term t, with the inverse of DF being IDF. Thus, the TF-IDF model primarily utilizes statistical principles to identify words that appear frequently in the current text but infrequently in other documents—words that can represent the document' s distinctive characteristics.

The advantages of the TFIDF algorithm are evident: first, its principle is simple and easy to implement; second, it comprehensively considers the situation of feature terms both within individual texts and across the text collection, making features selected through the dual TF and IDF process more representative.

4.2 KeyBERT Keyword Extraction

KeyBERT is a novel and straightforward keyword extraction technique whose principle is to utilize BERT embeddings to create keywords and key phrases most similar to the document. The ultimate goal of the BERT model is to use unlabeled corpora for training to obtain semantic information within texts —essentially, the semantic representation of text—which is then fine-tuned for specific NLP (Natural Language Processing) tasks and ultimately applied to those tasks. In NLP methods based on deep neural networks, words in text are typically represented using one-dimensional vectors (commonly called "word vectors"). Therefore, the core information input in the BERT model primarily refers to vectors of original words, which can be initialized or trained using algorithms similar to Word2vec. The information output mainly represents the semantic information contained in words within the text, using vector representations.

Keywords and documents share consistent semantic representations, and leveraging BERT' s encoding capability can achieve favorable results. However, the disadvantages are also apparent: first, different semantic encoding models produce different results; additionally, BERT can only accept text of limited length,

requiring preprocessing measures such as abstract extraction when handling long texts, which increases time complexity and reduces accuracy. Therefore, this paper applies the KeyBERT algorithm to keyword extraction from abstract texts, demonstrating certain pertinence and practicality.

4.3 Semantic Similarity Calculation

This paper approaches keyword extraction from different text perspectives, performing targeted processing on abstracts and full texts. Different adaptive algorithms are employed to extract distinct keyword sets, fusing multiple linguistic features as extraction criteria to comprehensively consider the impact of different features on keyword extraction. The core of the fusion processing algorithm proposed in this paper is semantic similarity calculation, which processes different keyword sets through similarity measures to form a new keyword set capable of serving as core terms and representing article content as labels.

The similarity processing procedure for keyword sets is similar to text similarity calculation. The word segmentation process has already been completed in the text preprocessing stage, and the extracted candidate keyword sets are already representative, allowing direct vectorization processing. Due to the current lack of high-coverage lexical knowledge bases, semantic vectors are adopted to capture lexical relationships. Word2vec can vectorize each word based on given corpus information during word vector training, optimizing internal training model mechanisms to achieve word embedding. As a mainstream word vectorization method, its primary models include the CBOW model and Skip-gram model.

For similarity calculation of vectorized words, this paper employs cosine similarity to compute correlations between word vectors. Cosine similarity uses the cosine value of the angle between two vectors in vector space as a measure of difference between two individuals. A cosine value closer to 1 indicates that the angle between two vectors approaches 0 degrees, meaning the vectors are more similar. Conversely, a cosine value closer to 0 indicates the angle approaches 90 degrees, meaning the vectors are less similar. After representing case texts as two vectors A and B through spatial vector construction, vector similarity can be calculated to measure the semantic similarity between two words.

$$similarity(A, B) = \cos(\theta) = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

5 Experiments

5.1 Text Preprocessing

Data Acquisition: To ensure smooth model construction and output accuracy, experimental data must select a scientific document containing abstract,

keywords, and full text sections. This experiment uses an article from the automation domain in scientific literature as an example. The main reasons for this selection are: first, the article contains numerous technical terms, facilitating learning training and semantic integration; second, the text has distinct thematic characteristics, making it easier to capture word vector features.

Text Preprocessing: Collected experimental texts are segmented, with preprocessing operations performed on the full text portion, including word segmentation, part-of-speech tagging, and stop-word filtering. The abstract portion does not require separate preprocessing (the KeyBERT model encapsulates preprocessing operations such as word segmentation).

Using a scientific literature article from the automation domain as an example (all subsequent experiments use this article), its main research content undergoes word segmentation and part-of-speech tagging processing, with results as follows:

[Figure 3: see original paper] Original sentence output results

After performing stop-word removal processing on the segmented scientific literature paragraphs, the results are as follows:

[Figure 4: see original paper] Word segmentation output results

[Figure 5: see original paper] Stop-word removal output results

The results demonstrate that using the jieba third-party segmentation library can filter out the vast majority of stop words and invalid words. These prepositions and conjunctions not only increase the workload of automatic indexing but also interfere with indexing accuracy. Through effective data cleaning, relatively clear data materials are obtained for subsequent text indexing use.

5.2 Keyword Extraction

The abstract portion already represents the key content of the text. Using the KeyBERT algorithm, the top N words by extraction weight are directly selected to form candidate keyword set A. The full text portion, after text preprocessing, generates a candidate word set. The TF-IDF algorithm then traverses this candidate set word by word, sorting each word node' s final score from high to low, and extracting the top N high-scoring words as candidate keyword set B.

Following the experimental approach, indexing term extraction experiments are conducted separately on this paper' s abstract and full text, with the candidate indexing term count threshold uniformly set to 10. The results are as follows:

Experiment 1 - Abstract KeyBERT processing results: Petri

Experiment 2 - Full text TF-IDF processing results: Petri token

Comparing the results of these different processing methods for different article structures, commonalities can be observed. This is because the abstract already

summarizes the full text, and words appearing in the abstract will likely be the themes of the full text, reappearing in the body to reflect the research focus. The former captures semantic features more precisely in summary-type short texts, while the latter captures word frequency features in the entire full text through TF-IDF. The necessity for subsequent fusion processing arises precisely because different candidate word sets exhibit commonalities and similarities.

5.3 Keyword Fusion Processing

Fusion processing is performed on the two generated keyword sets A and B to produce the final keyword set C as the ultimate keyword collection.

(1) Feature weight calculation for candidate terms.

Parameter Setting: Since this is a progressive numerical comparative experiment, parameter settings do not require rigorous experimental determination. Assigning fixed values does not affect experimental result comparisons, as long as the same parameters remain consistent throughout.

The indexing term results from KeyBERT for the abstract are recorded as set A, with corresponding weights denoted as Qa. The indexing term results from the improved TF-IDF for the full text are recorded as set B, with corresponding weights denoted as Qb. Fusion processing and weight calculation are shown in the following formulas:

$$A = \{a_1, a_2, a_3, ..., a_x\}$$
$$Qa = \{Qa_1, Qa_2, Qa_3, ..., Qa_x\}$$
$$B = \{b_1, b_2, b_3, ..., b_y\}$$
$$Qb = \{Qb_1, Qb_2, Qb_3, ..., Qb_y\}$$

In the above formulas, ax represents different KeyBERT candidate indexing terms, Qax represents corresponding weights, and x is the number of candidate indexing terms in set A; bx represents different improved TF-IDF candidate indexing terms, Qby represents corresponding weights, and y is the number of candidate indexing terms in set B. Qa and Qb have been normalized with value ranges of (0,1), enabling weight superposition processing.

The similarity fusion threshold value is α. Similarity between two words equal to or greater than α is defined as similar and can undergo fusion processing. Assuming set A indexing term count x is 4 and set B indexing term count y is 6, with indexing term a1 similar to b2, and indexing term a2 similar to both b5 and b6, the fused indexing term set C is as follows:

$$C = \{a_1, a_2, a_3, a_4, b_1, b_3, b_4\}$$

with corresponding weights:

$$Qc = \{Qa_1 + Qb_2, Qa_2 + Qb_5 + Qb_6, Qa_3, Qa_4, Qb_1, Qb_3, Qb_4\}$$

(2) Merging and Sorting.

Since set A consists of abstract-type indexing terms with inherently higher standardization, whose expressed meanings better reflect the main information of scientific literature, set A takes priority over set B during candidate term fusion, performing union processing. Partially overlapping candidate terms are merged with their weight values summed, and each candidate term's final value is sorted in descending order.

(3) Output Indexing Terms.

The top n words are extracted from the descending-order candidate terms as final keyword output. The most critical parameter is the similarity threshold α, which defines how large α must be to determine candidate term similarity. Comprehensive testing shows that $\alpha = 0.8$ yields good fusion effects. The indexing result sequence is limited to 5, meaning the top 5 words after fusion processing serve as the final indexing term set. The results are as follows:

Indexing results: PLC, PC, Petri net, control system, program design

The reference keyword set provided by the author is as follows.

Reference keyword set: PLC, Petri net, logic equation, control model, program design

5.4 Results Analysis

Using the keywords listed by the document author as reference keywords, analysis shows that the indexing model' s output results have high similarity with the reference keyword set, indicating good indexing effectiveness. Experimental results demonstrate that fusing two extracted keyword groups through the multi-feature fusion automatic indexing model proposed in this paper can produce relatively accurate output results. Therefore, the text automatic indexing method based on multi-feature fusion is feasible with relatively high indexing accuracy.

Based on the fundamental concept of multi-feature fusion, this paper proposes an automatic indexing model using the multi-feature fusion approach. The model consists of four components: input, candidate term extraction, candidate term fusion processing, and output. Candidate term extraction employs KeyBERT method and TF-IDF method to process abstracts and full texts respectively, extracting two groups of candidate keywords. The core technology in the candidate term fusion processing component uses cosine similarity calculation. This model not only combines the advantages of both algorithms to a certain extent but also comprehensively considers both accuracy and comprehensiveness in text indexing, providing a feasible approach for automatic indexing of key information in documents and possessing certain application value.

References

[1] Han Hongqi, Gui Jie, Zhang Yunliang, Weng Mengjuan, Xue Shan, Yue Lindong. Large-scale thesaurus automatic indexing method[J]. Journal of the China Society for Scientific and Technical Information, 2022, 41(05): 475-485.
[2] Yu Chun. Research progress in automatic indexing[J]. Library Science Research, 2012, (04): 18-22.
[3] Cai Yingchun, Zhao Xinru, Zhu Yumei, Wang Xiuxiu. Review and prospect of document indexing technology in China[J]. Library Journal, 2022, 41(03): 18-31.
[4] Zhang Jing. Review and prospect of automatic indexing technology[J]. Modern Information, 2009, 29(04): 221-225.
[5] Hans Peter Luhn. A Statistical Approach to Mechanized Encoding and Searching of Literary Information[J]. IBM Journal of Research and Development, 1957, 1(4).
[6] Zhao Pengmo, Dou Yongxiang et al. Information Resource Management Technology[M]. Beijing: Science Press. 2010.
[7] Wang Xiaolin. Improved TF-IDF keyword extraction method[J]. Computer Science and Application, 2013, (1): 64-68.
[8] Jiang Yi, Huang Yong, Xia Yikun, Li Pengcheng, Lu Wei. Academic text vocabulary function recognition—application in keyword automatic extraction[J]. Journal of the China Society for Scientific and Technical Information, 2021, 40(02): 152-162.
[9] Zhang Y, Xiao W. Keyphrase Generation Based on Deep Seq2seq Model[J]. IEEE ACCESS, 2018, 6: 46047-46057.
[10] Li Qianju, Li Sida, Liu Jianyi. A keyword automatic indexing method based on knowledge organization[J]. Information Science, 2016, 34(11): 107-110+139.
[11] Wang Xing, Liu Wei. Research on automatic indexing of Chinese academic literature based on citation[J]. Library and Information Service, 2014, 58(03): 106-110+105.
[12] Zhang Chengzhi, Hu Shaohu, Zhang Yingyi. Exploring the performance improvement of microblog keyword extraction by eye movement data from general corpora[J]. Journal of the China Society for Scientific and Technical Information, 2021, 40(04): 375-386.
[13] G. Salton, A. Wong, C. S. Yang. A vector space model for automatic indexing[J]. Communications of the ACM, 1975, 18(11).

Author Contributions: Ju Jiachen: Proposed research ideas, paper writing; Song Peiyan: Research framework design, paper revision

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