Berger Scientfic Press

Innovation & Technology Advances, 2025, 3(2), 19-45
https://bergersci.com/index.php/jta/index
DOI: 10.61187/ita.v3i2.222

A New Methodology for Chinese Term Extraction from Scientific Publications

Huaili Zheng*, Ting Jiang

School of Computer and Artificial Intelligence, Nanjing University of Finance & Economics, Nanjing, China

*Correspondence: zhenghuaili@nufe.edu.cn

Abstract: To identify Chinese technical terms, this study focuses on extracting terms from a corpus of scientific publications. The process begins with the identification of term boundaries, followed by the application of Chinese part-of-speech (POS) patterns to extract candidate terms. Features of words or characters that signal term boundaries are defined, enabling the segmentation of sentences into smaller units and facilitating the removal of irrelevant terms that may not be filtered by other approaches. POS patterns are specifically designed for the extraction of Chinese technical terms. A comparison between candidate terms extracted using these POS patterns and those obtained via n-gram models shows that the proposed POS-based method effectively eliminates a significant portion of non-relevant terms while retaining most useful ones. In the term scoring phase, a novel method based on contextual information—referred to as the Hellinger distance for context information acquisition—is introduced. This approach proves more effective than existing context-based methods. Subsequently, the Hellinger distance method is integrated with Kullback–Leibler divergence to evaluate terms along the dimensions of informativeness and phraseness. The proposed term scoring method is compared with eight alternative approaches. Results demonstrate that it outperforms others in scoring Chinese terms, particularly in the extraction of multi-word terms.

Keywords: Automatic term extraction; Technical term extraction; Terminology extraction; Context information; Chinese term extraction.

How to cite this paper: Zheng, H., & Jiang, T. (2025). A new methodology for Chinese term extraction from scientific publications. *Innovation & Technology Advances*, 3(2), 19–45. https://doi.org/10.61187/ita.v3i2.222

1. Introduction

Technical terminologies are single words or multi-word expressions that denote specific concepts within a specialized domain. They play a crucial role in ontology development, lexicon enhancement, and query expansion in information retrieval systems. Scientific publications represent a key source of such terminologies, offering a high-quality corpus rich in technical terms—including emerging terminology. Compared to English, Chinese term extraction remains a less explored area. This study aims to extract Chinese technical terms from scientific publications without restricting the domain, thereby accommodating interdisciplinary research and enhancing the extensibility of the proposed method.

Manual term extraction is time-consuming and labor-intensive, making automated approaches a valuable alternative. Automatic Term Extraction (ATE), also referred to as Automatic Term Recognition (ATR), is the task of identifying domain-specific terminologies from technical corpora through computational means. Although ATE has been studied extensively for decades, limited attention has been given to Chinese technical term extraction. Key challenges in this area include: (1) The inapplicability of certain English part-of-speech (POS) patterns and filtering rules due to structural differences between Chinese and English terms; (2) A predominance of research focused on biomedical domains, where corpus structure and terminological characteristics may differ significantly from those in other scientific fields. Typically, term recognition is treated as a two-stage process: candidate extraction and term scoring. The first stage identifies potential terms using linguistic patterns, n-gram models, and filtering mechanisms such as stop-word lists. The second stage assigns a relevance score to each candidate term, facilitating the selection of the most appropriate domain-specific terms.

In candidate extraction, recent approaches often employ POS-based linguistic filters or n-grams supplemented with stop-lists to eliminate unsuitable candidates. However, several issues persist when applying these methods to Chinese: (1) English POS patterns are not directly transferable due to structural differences in terminology; (2) The absence of word delimiters in Chinese necessitates word segmentation and POS tagging, which can introduce errors due to fault propagation; (3) The n-gram approach coupled with filtering rules tends to yield substantial noise; (4) Stop-lists are often manually constructed and domain-specific, limiting adaptability across domains.

In the term scoring phase, most prior work has relied on statistical measures based on either informativeness or phraseness, with few methods integrating both aspects. Additionally, although contextual information is recognized as valuable for term extraction, its standalone use is often inefficient, and thus it remains underutilized. To address these issues, this study employs unsupervised methods in both the candidate extraction and term scoring phases to minimize manual intervention and improve cross-domain applicability. During candidate extraction, we introduce a boundary recognition method to identify term borders, enabling more accurate segmentation of text. We also develop specialized POS patterns for Chinese candidate term identification. In the scoring phase, we propose a novel approach that leverages contextual information and combines both informativeness and phraseness into a unified scoring framework.

The main contributions of this paper are as follows: (1) A statistical method for detecting term boundaries to improve phrase selection; (2) Design of POS patterns tailored for Chinese candidate term extraction; (3) An unsupervised scoring method based on contextual information to evaluate term integrity; (4) A unified scoring model that integrates both informativeness and phraseness to enhance extraction performance.

2. Related Work

Recent studies on terminology extraction have addressed a variety of approaches by using supervised or unsupervised techniques. They can be divided into three categories: (1) linguistic, (2) statistical, (3) machine learning, and (4) hybrid. In this section, we discuss the characteristics of those methods to find appropriate measures to extract good terminologies.

2.1. Linguistic Approaches

Linguistic approaches attempt to extract terms by using linguistic patterns from learning specific syntactical structures of terms. These techniques are often called shallow linguistic filtering techniques, which are often implemented as part-of-speech (POS) filters or phrase chunking. Since the 1990s, candidate terms have been extracted from noun phrases [1]. After that, Justeson and Katz accept terms as noun sequences containing adjectives, nouns, and occasionally prepositions and propose that the terms are not only single-word phrases but also multiword phrases [2].

Linguistic techniques are often applied in candidate term extraction. Although linguistic patterns may yield useless terms and omissions, well-defined patterns also improve the extraction accuracy. The choice of the linguistic filter affects the precision and recall of the final term list. Thus, an important issue is how to define appropriate patterns. Chinese are written without using spaces or other word delimiters. Before the use of linguistic patterns, we need to divide the corpus into small segments (the process is called word segmentation). The POS tags of these segments are hard to define; sometimes, they may vary depending on the context. Therefore, linguistic patterns of Chinese term extraction are different from those predefined patterns in previous studies.

2.2. *Statistical Approaches*

Statistical approaches usually aim at assigning a score to the candidates according to a criterion to rank terms. A higher score in the output list indicates a more relevant term.

Frequency is the most widely used measure in term extraction. The more often a term occurs in a domain corpus, the more relevant it is to the domain. According to Verberne, the frequency can be either implemented as a raw term count or as the maximum likelihood estimate of the probability of occurrence of a term in the domain, such as the term frequency [3]. Unfortunately, the top lists of the output phrases obtained via the term frequency method are usually generic phrases. Term frequency inverse document frequency (tf-idf) is a famous method developed for text retrieval; it considers term specificity as a weight to lower the score of generic phrases that occur in more documents [4]. It is often used as a baseline in comparison with other term scoring methods.

Most statistical approaches extend the frequency criterion with either of two principles: informativeness or phraseness.

Informativeness expresses how much information a term can provide about the domain collection. Most studies extract informative terms from a comparison of the domain corpus with the background corpus. The C value considers all the following characteristics of the candidate terms: the frequency of the candidates, the frequency of the other candidates that have the current candidates as substring, the total number of these other candidates, and the length of the candidate term [5]. They also consider the context information in the N value part. They use the context words of the top-N output terms via the C value measure as a base context word set. Then, the weighted sum of the co-occurrence frequency of the candidate and the context words is computed.

Phraseness expresses the strength of the combination of words in multiword phrases. Mutual information is necessary to measure the associations between words [6]. However, the method overestimates collocations composed of low-frequency words. To alleviate this problem, Pantel and Lin use log-likelihood, which is more robust to low-frequency events to compensate [7]. They combine both methods in extracting multiword terms. Cooccurrence-based chi-square is another method for scoring the phraseness of terms [8]. The cooccurrences between each term and the most frequent terms in the corpus are considered in the method.

Tomokiyo and Hurst use pointwise Kullback–Leibler divergence for scoring both informativeness and phraseness. The informativeness score is computed using a KL-divergence model that compares a foreground corpus against a background corpus, quantifying the information loss when approximating the phrase distribution with the background model [9]. Verberne subsequently improved the method by adding a parameter relative to the phraseness component to determine the proportion of multiword terms in the output [3].

2.3. *Machine Learning Methods*

Machine learning methods focus mainly on learning features from training data for the purpose of term extraction. It can be classified into three categories: supervised learning, semi-supervised learning and unsupervised learning.

The most frequently used high-performance measure is supervised learning. Many methods have already been used in term extraction tasks, such as naïve Bayes, SVM, and CRF. Kovačević used conditional random fields (CRFs) to accomplish the automatic terminology extraction task. They combine both linguistic and statistical (based on frequency) features in feature selection [10]. The performance of supervised learning relies heavily on the quality and scale of the training set, which is annotated. Most of these studies are based on domain-specific knowledge such as lexicons or ontologies in the domain. However, most of the domains do not have domain-specific knowledge and need manual annotation, which is an arduous task.

Many unsupervised measures have also been applied in term extraction. Judea et al. used two methods (a term candidate classifier and a conditional random field) to extract terminology from patents [11]. Because those methods train on automatically labeled training data, their measures are unsupervised.

Feature selection has a strong influence on the performance of machine learning methods. The feature set always obtains statistical (e.g., TF, IDF), linguistic (e.g., POS tags), and hybrid knowledge (e.g., C value) from the training corpus. da Silva Conrado et al. conducted experiments on unigram extraction via machine learning with different feature sets [12].

2.4. Hybrid Methods

To achieve better performance, combinations of these methods have been conducted in many studies. Most of those methods are based on a combination of linguistic and statistical measures.

Lossio-Ventura et al. combine two measures to extract multiword terms. The first is the LIDF value (lingistic patterns, IDF, and C value information), which is both a linguistic and a statistical-based measure. The second is TeRGraph (Terminology Ranking Based on Graph Information), which is a graph-based measure (statistical). The graph-based measure assumes that a term with more neighbors is less representative of the specific domain and uses the Dice coefficient to compute the co-occurrence between two terms connected by the edge of the graph [13]. After the method was proposed, Lossio-Ventura et al. combined the method above with a new web ranking measure, WAHI (web association based on hits information), in the term ranking stage. The method uses search engines to measure word associations [14]. Ittoo and Bouma combine both linguistic and statistical methods to extract multiword terms (primarily designed to identify terms with 2 words). They use some linguistic filtering measures in the first step. In terms of the ranking phase, they use the cube mutual information (MI3) measure with the English Wikipedia collection to compensate for the unavailability of domain-specific knowledge resources [15].

There are also other combinations of measures for term extraction. Bolshakova et al. conducted experiments to compare topic models (based on k-means, spherical k-means, hierarchical agglomerative with single, complete and average linkages, and NMF) applied to the task of single-word term extraction and choose LDA among probabilistic topic models. They reveal the topics in the collection at the first stage and then use statistical methods based on frequency to rank the terms [16].

Although the linguistic filtering method does not need to extract all types of terms, such as, all n-grams (unigrams, bigrams, etc.) are extracted, which results in too much noise (useless phrases) and makes it difficult for the statistical methods used in the next step to extract real terms from those phrases [17]. There are still too many useless phrases left even when the output terms with a stop list are filtered. Wermter and Hahn conducted a contrastive study of purely statistics-based measures and frequency methods and reported that the statistical method results in virtually no difference compared with the frequency of occurrence counts, whereas linguistic methods can result in a marked difference in term extraction [18]. Therefore, we use the linguistic approach with the term border identification method for candidate term extraction.

3. Method

The methodology for Chinese technical terminology extraction consists of two steps:

- Candidate term extraction;
- Ranking of candidate terms.

3.1. Candidate Term Extraction

To generate candidate terms from the Chinese domain corpus, we first split the domain corpus into small segments according to the words indicating the term borders. Therefore, we use an unsupervised statistical method to identify the borders of terms, which is referred to as term border recognition. Second, we apply filtering for part-of-speech patterns to extract candidates between borders. We construct the Chinese POS pattern to extract the candidates. After that, we filter out some candidate terms (such as the candidate term that only appears in one document or the candidate term that is substring of another candidate term but tends to appear at a similar frequency in the corpus) via simple processing methods.

3.1.1. Term border recognition

Documents, from which candidates are generated, contain words indicating the borders of terms so that terms will never cross (such as punctuations, prepositions and conjunctions). Most likely, they can be either general words not associated with any domain or special words with low frequency (such as idioms, archaism or named entities such as names). In this paper, those words are named term border words.

Most studies use a generic stop list to filter candidates extracted by POS patterns or n-grams, and a well-defined stop list greatly improves precision. Since the stop lists are manually generated and domain dependent, they are difficult to use for filtering candidates in another domain.

Yang et al. focused on extracting words that indicate the borders of terms. They call these words delimiters. First, on the basis of a stop-list or domain lexicon, they take the predecessors (the words before the term) and successors (the words after the term) of the term in the lexicon as delimiter candidates, rank the candidates by frequency and extract the top-N candidates as delimiters. The domain lexicon may not be available in the domain, and a stop-list is also not feasible. The performance of the method depends on the size of the domain lexicon; a larger domain lexicon means that more comprehensive candidate delimiters are extracted, resulting in a more accurate delimiter set. In addition, they extracted delimiter words on the basis of frequency, which means that high-frequency domain words are combined and low-frequency words are omitted. The result contradicts the hypothesis that delimiters are domain independent [19].

The term border words are similar to stop-lists. Unlike the manually collected stop list and method, these words have some features that can be distinguished from other words, which can be generated in an unsupervised way [19]. Therefore, we first define the features of these words.

(1) General words: domain-independent and high-frequency words

- Common used characters: Common used characters includes punctuations, Arabic numerals.
- Common used words: Common used words can be single characters or words consist of more than one characters (In Chinese language, sentences can be splitting to words after word segmentation, each word consists of one or more characters). They are high frequency words in Chinese language, such as "一" (one), "和" (and), "是" (is), "主要" (main) etc.

(2) Special words with low frequency

- Named entities: Phrases such as person name, place name and organization name.
- Special characters: Characters such as mathematical notation, unit symbols, Greek letters, etc.
- Rarely used Chinese characters.
- Idioms: Idioms include Chinese four-character idiom and idiolects. Idiolects are words comes from personal writing habits or language habits of the author.
- Archaism in classical Chinese.
- Digital gibberish or Chinese garbled: Those characters often derive from a format transformation process such as casting PDF document to TXT document.
- Wrongly written or mispronounced characters.

- Domain specific terms from other domain: Such as biology terms for information science domain.

Term border recognition can be divided into three steps:

(1) Words segmentation

Word segmentation is commonly used in Chinese NLP tasks because delimiters do not exist between words in the Chinese language. First, we use a Chinese tokenizer to split the domain corpus into small segments, which refer to words in this paper.

(2) Low-frequency word extraction on the basis of frequency

As described above, special words usually have two characteristics: low frequency in a domain, such as domain-specific terms from other domains, or low document frequency in a domain because of the personal habits of the author. Therefore, we combine two measures to extract special words with low frequency.

Term frequency (TF) is the standard notion of frequency in natural language processing; it counts the number of times that a term/word appears in a corpus. In this paper, we rank the words by term frequency after segmentation, then use the last-k words as low-frequency words and add them to the term border word list. The term frequency can be expressed as follows:

$$tf(t,D) = \frac{count(t,D)}{|D|} \tag{1}$$

$count(t,D)$ is the number of times that term appears in collection D. $|D|$ is the size of D, which represents the total number of words in D.

The document frequency (DF) is derived from the information retrieval domain and considers the number of documents that contain a term/word. Document frequency is always converted into inverse document frequency (IDF) in document collection as an important term weighting method. In this section, we need not consider the inverse document frequency. Since some low-frequency words come from the personal habit of the author, or few cross-domain articles that the domain rarely mentions, we need to obtain only those words that are contained in few documents. Therefore, we count the number of documents that contain a term/word as the document frequency.

(3) Extraction of domain-independent words with high frequency on the basis of Kullback–Leibler divergence and Chinese Wikipedia

The primary task in this step is to extract general words with high frequency but not related to any domain. High-frequency words can be extracted on the basis of frequency, but in this way, domain-related words with high frequency are also extracted; however, we cannot put high-frequency domain-related words into the removed list because they are important parts of the domain-related terms. To address this problem, we use Chinese Wikipedia as an external document corpus for comparison with the domain corpus. Chinese Wikipedia is an article collection that contains many articles, and it has been successfully used as an external resource in many natural language processing tasks. Terms that are domain related appear more frequently in the domain corpus than in the general external corpus. Therefore, a measure is needed to calculate distances of word frequency between these two domains.

Kullback–Leibler (KL) divergence is a measure that defines the difference between two probability distributions. In (Tomokiyo and Hurst 2003), pointwise KL divergence was used for keyphrase extraction. In this step, KL divergence is used to estimate the loss of information between two domain corpora D (domain corpus) and E (external corpus), which is the contribution of the word to the expected loss of the entire distribution. The KL divergence between two probability mass functions is defined as:

$$KLdiv(w) = P(w|E)log\frac{P(w|E)}{P(w|D)} \tag{2}$$

where w is one of the words from the domain corpus, $P(w|D)$ is the probability of word w in domain corpus D, and $P(w|E)$ is the probability of w in the external corpus E (Chinese Wikipedia). $P(w|D)$ is estimated as the relative term frequency of w in D. Some words may not occur in the external corpus, and we estimate $P(w|E)$ as $1/|E|$, in which $|E|$ is the size of the external corpus.

By applying the Kullback–Leibler (KL) divergence measure to arrange each word with a score in the domain corpus, we extract the top-N words in the output list as high-frequency domain independent words.

3.1.2. Chinese POS filtering for candidate term extraction

Linguistic patterns are commonly used in candidate term extraction. Well-defined POS patterns can improve the precision of term extraction but also exclude some terms with special syntactic structures that are not present in predefined patterns. As described in Sec. 1, the Chinese language is different from English; if we apply only the normal syntactic patterns designed for English, we will lose too many terms. Therefore, we built a list of linguistic patterns according to the syntactic structure of terms present in Chinese scientific articles.

To analyze the syntactic structure of Chinese terms, we need a preexisting Chinese term list with POS tagging. Because we aim at extracting technological terminology, we collected a keyword set from scientific articles listed by the authors. After removing some keywords with special characteristics (such as punctuations, full-width English characters, graphic symbols, etc.), we obtained 8473 terms to construct the patterns. Then, we choose NLPIR [20] to address POS tagging.

In general, instead of using commonly used patterns, studies usually select the top-N high-frequency patterns by computing the frequency of the syntactic structures of terms present in the corpus. However, the diversity of Chinese phrases leads to many POS tag combinations, and the patterns change according to the term set. It is difficult to define the number of high-frequency patterns to extract the candidates, and most of the low-frequency patterns are missing. Therefore, we carry out a new POS pattern construction method for Chinese. We take the POS pattern construction of single-word terms and multiword terms as two different tasks. For single-word terms, we select the top-k frequent POS tags as patterns. For multiword terms, we compute the frequency of POS tags for each word contained in different positions (first word, last word and intermediate words) of the terms; then, we compute the frequency of the syntactic structures of each 2-gram inside the terms. Patterns among the top-k frequencies are selected to build the list of patterns for multiword terms.

After POS pattern construction, we apply the POS patterns to the whole corpus to extract the candidate term.

3.1.3. Filtering out some special candidate terms

Some special candidate terms, such as the low-frequency term, appear only once in the domain corpus or appear in one document, and terms that are substrings of other candidate terms (parent terms) tend to have a similar frequency in the corpus. Therefore, we do not need to extract those substrings. We carry out the filtering process via several measures.

For low-frequency term extraction, in an extreme case, the term only appears once or appears in one document of the domain corpus. Because the term only appears once in the domain corpus, it is difficult to define whether the candidate term is a true term or not, and it is more difficult to judge whether it is a domain-related term. Moreover, most of these words are useless terms. For terms that only appear in one document, the following are some cases: idiolects that come from the author's personal writing habits or language habits; specific terms from other domains; and low-frequency terms. Most of the terms in the abovementioned situation are not domain-related terms. Therefore, we filter out these low-frequency words.

Candidate terms may be extracted with their substring terms via POS patterns; thus, this article refers to these terms as parent terms. If the frequency of the substring term and its parent term are the same or similar, we filter out the substring term and keep only the parent term. To filter out the terms described above, we define the ratio of the frequency of the parent term to the frequency of the substring term and set a threshold value to filter out the substring. The ratio is computed as follows:

$$ratio = \frac{tf(t_p, D)}{tf(t, D)} \tag{3}$$

where t is a candidate term and where t_p is a candidate term in the candidate term set, which has t as a substring. $tf(t, D)$ is the term frequency of term t.

3.2. Term Ranking

In the term ranking stage. We consider both the informativeness criterion and the phraseness criterion for the ranking terms. We propose a new measure based on the context information of terms named the Hellinger distance for context information and combine the informativeness and phraseness of terms to arrange each term into a score. Finally, the top-N of the ranked list is output.

3.2.1. Hellinger distance for context information acquisition

Although there is a wide consensus that context information is useful for term extraction, few studies focus on context information during the process. The most relevant method is the NC value [5], which uses context information by combining both linguistic and statistical information. Linguistic information is in the form of a syntactic filter that restricts context words to nouns, adjectives and verbs. Statistical information assigns each context word a weight term on the basis of the frequency with which it appears with terms.

Context words are words that either precede or follow the candidate terms. Unlike the NC value, we do not restrict context words to nouns, adjectives or verbs, and the syntactic structure of context words can be any kind. Context words, not only nouns but also adjectives and verbs, provide clues for extracting terms; for example, if a candidate term follows with a punctuation, then it is more likely to be a complete term.

Hypothesis: We assume that the context words of all the candidate terms constitute a standard context word list. Our hypothesis is that if the context word distribution of the candidate term is similar to that of the standard context word list, then the candidate term is more likely to be a true term.

In this section, we propose a measure to acquire context information for term extraction, which refers to the Hellinger distance for context information (HDCI) in this paper. We describe the steps taken in the measure to construct a list of candidate terms from a corpus. First, we need to generate a standard context word list from the context of the candidate terms. Second, we compare the context words of the candidate terms with the standard context words.

Step 1. Generate a standard context word list

In this step, we need to define the standard context word list. We assume that the standard context word list is made up of some words that appear in the context of candidate terms. Therefore, there are two ways to generate standard context words:

(1) Extracting the context words from the context of ”important” candidate terms.

The NC value uses a set of terms extracted from the corpus to define the context weighting factor, which is called “important” term context words. The special terms are extracted from candidate terms, which are ranked according to their importance by other term ranking methods (C values). By defining the threshold of the scores, we obtain a set of “important” terms. Then, the standard context words are extracted from the context of the “important” terms.

(2) Generate all the words from the context of the candidate terms and generate the standard context word list by filtering out some low-frequency words.

Context words, which frequently cooccur with specific terms, indicate that they may consist of a fixed term. However, we do not know the influence of low-frequency context words on term extraction; therefore, we carry out an experiment to study the effect of low-frequency words. To obtain a standard context word list, we obtain that all the words precede or follow the candidate terms and then filter out some low-frequency words as standard context words.

To verify the necessity of generating a standard context word list via the two methods above, we perform some experimental research in Sec. 4.4.1.

Step 2. Term ranking-based Hellinger distance between the context word list and context words of the candidate term

Context words contain words that precede or follow the terms. In this step, we calculate the context score of the preceding word set and the following word set separately and combine both scores. Our assumption is that the more similar the distribution of the context words between two sets is, the greater the likelihood that the candidate term is an important term.

The Hellinger distance (HD) is a measure of distributional divergence, which is a well-established metric for calculating the distance between probability distributions [21]. The Kullback–Leibler divergence as well as the χ^2 measure and the Hellinger distance are particular cases of the family of f–divergences [22]. The widely used KL divergence and the χ^2 measure are not strictly distance metrics, which makes the Hellinger distance very appealing for our purpose of measuring the distance between two context word sets.

Unlike KL divergence for term extraction, the Hellinger distance focuses on measuring the distance of the probability between two domains represented by the terms, whereas KL divergence is used to measure the loss of information between two domains represented by the terms. Therefore, we use the Hellinger distance to determine the difference between the two probability distributions. The higher the score is, the greater the difference between the standard context word list and the context words of the candidate term. Both the scores previous and next to the term are calculated as:

$$\mathrm{HDCI}(t) = \sum_{w_i \in C_s} (\sqrt{P(w_i|C_s)} - \sqrt{P(w_i|C_t)})^2 \tag{4}$$

where t represents the candidate terms in the domain corpus, C_s represents the set of standard context words, and C_t represents the context words of the candidate term t. w_i represents the words from standard context words. $P(w_i|C_t)$ is computed as follows:

$$P(w_i|C_t) = \frac{count(w_i)}{\sum_{w_j \in C_s \cap C_t} count(w_j)} \tag{5}$$

where $count(w_i)$ is the number of words w_i in the context words from candidate term t and where w_j is from the context words of term t and is contained in the standard context word list C_s. If w_i is not contained in the set C_t, then we set $P(w_i|C_t) = 0$.

We combine the scores of the word set before or after the terms as follows:

$$\mathrm{HDCI}_{all}(t) = \mathrm{HDCI}_{before}(t) + \mathrm{HDCI}_{after}(t) \tag{6}$$

where $HDCI_{before}(t)$ is the score computed from the previous context word set of the term. $HDCI_{after}(t)$ is the score computed from the context words next to the term.

3.2.2 Methods for scoring the terms

In this section, we describe several methods that we use or evaluate in this paper. These methods are the Kullback–Leibler divergence for informativeness and phraseness, the C value/NC value and the left/right branching entropy.

(1) Kullback–Leibler divergence for informativeness and phraseness (KLIP)

Kullback–Leibler divergence for informativeness and phraseness has both an informativeness component and a phraseness component:

$$\mathrm{KLIP}(t) = \mathrm{KLI}(t) + \mathrm{KLP}(t) \tag{7}$$

Kullback–Leibler divergence for informativeness (KLI) is used to estimate the distance of the probability distribution of terms in two domain corpora, D (domain corpus) and E (external corpus). KL divergence is used to measure the loss of information between two domains represented by the term. It is defined as:

$$\mathrm{KLI}(t) = P(t|D)log_2\frac{P(t|D)}{P(t|E)} \tag{8}$$

where t is one of the candidate terms from the domain corpus, $P(t|D)$ is the probability of the term t in domain corpus D, and $P(t|E)$ is the probability of t in the external corpus E (Chinese Wikipedia). $P(t|D)$ is estimated as the relative term frequency of t in D. Since terms in corpus D may not occur in the external corpus E, we estimate $P(t|E)$ as $1/|E|$, in which $|E|$ is the number of words included in the external corpus E.

Kullback–Leibler divergence for phraseness (KLP) is designed for multiword term extraction and estimates the loss of information by assuming the independence of each word by applying the unigram model instead of the n-gram model. KLP is defined as:

$$\mathrm{KLP}(t) = P(t|D)log_2\frac{P(t|D)}{\prod_{i=1}^{n} P(u_i|D)} \tag{9}$$

where u_i is the ith unigram inside the n-gram t and where $P(u_i|D)$ is the probability of the unigram in D. Because $KLP(t)$ is given a score of 0 when it is used to compute the score of single-word terms, it is invalid for scoring single-word terms.

(2) C value/NC value

The C-value method is a combination method of linguistic and statistical information. Here, we describe the statistical part; the C value considers the importance of both the frequency and length of each candidate term, and it also takes the subset (extracted from the set of candidate terms) that contains the candidate term as substring into consideration. The greater the number and higher the frequency of the candidates in the subset are, the lower the score of the candidate term.

$$\text{C-value}(t) = \begin{cases} log_2|t| \times count(t,D) & \text{if } S_t = \phi \\ log_2|t| \times (count(t,D) - \frac{1}{|S_t|} \times \sum_{t' \in S_t} count(t',D)) & \text{if } S_t \neq \phi \end{cases} \tag{10}$$

where t is the candidate term, $|t|$ is the number of words in t, and $count(t,D)$ is the frequency of t in the unique document D. S_t is the set of terms that have t as substring, and $|S_t|$ is the number of terms in S_t. Because the C value was designed for the recognition of multiword terms, single-word terms are given a score of 0. To avoid the 0-score for single-word terms, we modified $log_2|t|$ to be $g_2(|t|+1)$.

The NC value is a hybrid method in which the N value part uses context information to extract the term.

$$\text{NC-value}(t) = \alpha \cdot \text{C-value}(t) + \beta \cdot \text{N-value}(t) \tag{11}$$

where t is the candidate term. The two factors of the NC value, the C value and the context information factor, according to the original paper, are assigned weights of 0.8 and 0.2, respectively. The N value is calculated as follows:

$$\text{N-value}(t) = \sum_{w_i \in C_t} \frac{count(t,w_i)}{count(t)} \times weight(w_i) \tag{12}$$

where w is the context word (noun, verb or adjective) to be assigned a weight as a term context word and where $weight(w)$ is the assigned weight to the word w. C_t is the set of distinct context words of t. w_i is a word from C_t.

$$weight(w) = \frac{\sum_{t_j \in D} count(t_j, w)}{N} \tag{13}$$

where t_j is the term in the domain corpus with which w appears, $count(t_j, w)$ is the number of terms t_j with which word w appears, and N is the total number of terms considered.

(3) Left/right branching entropy

In addition to the C value, another term scoring method that considers the context information of terms is left/right branching entropy [23]. It is defined as:

$$\text{L/R entropy} = \sum_{w_i \in C_t} p(t, w_i) \times log_2 p(t, w_i) \tag{14}$$

where t is the candidate term, w_i is one of the context words (words before or after the term), C_t is the context word set of term t (C_t is the context words before term t when computing the left branching entropy), and $p(t, w_i)$ is the probability of having w_i given t, which is computed as follows:

$$p(t, w_i) = \frac{count(t, w_i)}{count(t)} \tag{15}$$

where $count(t, w_i)$ is the frequency count of the co-occurrence of the candidate term t with the context word w_i and where $count(t)$ is the total frequency count of the candidate term t in the domain corpus.

3.2.3 Combining informativeness and phraseness for term ranking

To arrange each candidate score, we propose a new method to combine both informativeness and phraseness.

The term scoring methods described above focus on different aspects of the terms. KLIP has both an informativeness and a phraseness component. The C value is a method for scoring the phraseness of terms that considers the length and frequency of the candidate terms, as well as the length and frequency of the subset of the candidates that have the candidate term as substring. On the basis of the C value, the NC value also considers the context information of the terms.

Here, we propose a new method for scoring both the informativeness and phraseness of terms. It is defined as:

$$\text{KLIP-HD}(t) = \frac{\text{KLIP}(t)}{P(t|D) \times \text{HDCI}(t)} \tag{16}$$

4. Experiments and Results

4.1. Data, Preprocessing and Evaluation Methods

4.1.1. Data

We use two corpora for our experiments. The first is a domain corpus extracted from the Chinese scientific journal "Journal of the China Society for Scientific and Technical Information". The corpus consists of 250 Chinese-language scientific articles with a total of 4291024 Chinese characteristics. The second corpus is Chinese Wikipedia, which is made up of approximately 922,594 entries (until UTC 21:05 1-28-2025).

4.1.2 Preprocessing

The document corpus is processed by converting each document (from PDF) to plain text via ABBYY FineReader 12. Then, the texts are split into tokens by word segmentation with a total of 1893248 words, and the word segmentation and part-of-speech (POS) tags of the tokens (e.g., words) from the texts are obtained via NLPIR [20].

4.1.3 Evaluation method

For the evaluation, we create a term list generated by two human judges (annotators), who are well versed in the domain. Terms that both annotators annotated were marked as correct terms [24]. The terms can be single-word or multiword terms, and a total of 3417 terms are extracted from the 250-domain corpus manually.

The terms that are extracted and corrected are true_positives, and the terms that are extracted but incorrect are false_positives. The terms in the manual list that failed to extract are true_negatives. The precision scores of the terms extracted at the different threshold values are computed via Eq. 17. The recall scores for the different threshold values are then computed via Eq. 18.

$$Precision = \frac{\text{true\_positives}}{\text{true\_positives} + \text{false\_positives}} \tag{17}$$

$$Recall = \frac{\text{true\_positives}}{\text{true\_positives} + \text{true\_negatives}} \tag{18}$$

To obtain a single performance value, we determine the F1 score, which is computed via Eq. 19.

$$\text{F1} = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{19}$$

For the term scoring methods, we divide the candidate terms into four parts according to the ranking scores of the terms. The top 25% segment contains the candidate terms whose scores are among the top 25% (1 or top 25%), and the terms whose scores are between the top 25% and the top 50% are divided into the second part (2 or top 25%-50%), the third part (3 or top 50%-75%) and the fourth part (4 or top 75%-100%). The purpose of the term scoring method is to assign a high score to the true terms (denote domain-related terms and useful terms). In this evaluation method, the closer the number of true terms is to the top, the better the term scoring method is.

4.2. Evaluation of Term Border Recognition

In this section, we first define the threshold values of high-frequency words and low-frequency words as term border words. We then evaluate the influence of term border recognition on term extraction.

4.2.1. Defining the threshold of the term border word extraction

To extract the term border words, we define the threshold of the frequency. As described above, term border words can be low-frequency words or domain-independent high-frequency words.

For low-frequency word extraction, a threshold is selected to restrict the frequency of the words, and we choose the document frequency (DF) and term frequency of the words (TF) to define the frequency of the words. In the experimental process, we set $DF(w) < \{3,4,5,6\}$, and $TF(w) < \{10\%, 60\%, 70\%, 80\%, 85\%, 90\%\}$.

For high-frequency domain-independent word extraction, we use Kullback–Leibler (KL) divergence to define the difference between two word distributions. We rank the result of KLdiv and then choose the top-k most frequent words as the term border words. We choose the top {10%, 20%} of the scored words as the term border words in the experiment.

To define the threshold to extract the term border words, we need to evaluate the result of the candidate extraction. Therefore, the result is based on the candidate extraction.

The following is the process of candidate extraction for the experiment. After term border extraction, we apply POS patterns for linguistic filtering to obtain the candidate terms and filter out some special candidate terms. After that, we evaluate the result of the candidate term extraction. **Table 1** presents the results of candidate term extraction by defining different thresholds to extract the term border words.

Table 1. Defining the threshold of the term border word extraction

		Precision	Recall	F1
Before term border recognition		13.08%	90.64%	0.2286
Removing high frequency words	**top-10%**	**16.11%**	**87.94%**	**0.2724**
	top-15%	16.64%	87.09%	0.2795
	top-20%	16.93%	85.92%	0.2829
	top-25%	17.18%	85.13%	0.2859
	top-30%	17.45%	84.81%	0.2894
Removing low frequency words	DF <3, TF <0.1	13.24%	90.58%	0.2310
	DF <3, TF <0.6	13.24%	90.58%	0.2310
	DF <3, TF <0.7	13.33%	90.55%	0.2324
	DF <3, TF <0.8	13.90%	89.58%	0.2406
	DF <3, TF <0.85	14.29%	88.12%	0.2459
	DF <3, TF <0.9	14.79%	83.85%	0.2514
	DF <3, TF <0.95	15.44%	69.53%	0.2527
	DF <4, TF <0.85	14.30%	88.06%	0.2461
	DF <5, TF <0.85	**14.31%**	**88.06%**	**0.2462**
	DF <6, TF <0.85	14.28%	87.56%	0.2456
Removing high & low frequency words	top-10%, DF <5, TF <0.85	17.62%	85.48%	0.2922
	top-15%, DF <5, TF <0.85	18.10%	84.75%	0.2983
	top-20%, DF <5, TF <0.85	**18.32%**	**83.81%**	**0.3006**
	top-25%, DF <5, TF <0.85	18.13%	84.26%	0.2984

To reach a balance between precision and recall. For high-frequency domain independent word extraction, we set the scored words to the top 20% of the scored words. For low-frequency word extraction, we set DF <5 and TF <85% of ranked words. In the corpus, low-frequency words make up a very large proportion of all words. Therefore, the removal of low-frequency words can improve the precision of term extraction and simplify the process of the following steps. Moreover, the removal of high-frequency domain independent words also reduces the number of useless candidate terms. After term border extraction, it consumes less processing time and improves the precision of term extraction by removing the low-frequency words and high-frequency domain independent words. The following evaluation tasks are based on the thresholds we set in this step.

4.2.2. Evaluation of term border recognition

An experimental and comparative method is used to study the effect of term border recognition on term extraction. We use nine term extraction methods for term extraction to evaluate the effect of term border recognition. In addition to the methods we describe above, the experiments are also based on FP [25] and CB [26]. The results are shown in **Table 2**.

Table 2. Effects of term border recognition via nine methods for term extraction.

		Top 25%			Top 25%-50%			Top 50%-75%			Top 75%-100%		
		Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Before term border recognition	FP	27.11%	38.10%	0.3168	17.48%	25.43%	0.2072	11.59%	22.27%	0.1525	2.96%	2.49%	0.0270
	CB	27.61%	39.57%	0.3253	14.90%	21.25%	0.1751	10.62%	16.15%	0.1281	9.12%	11.33%	0.1011
	KLI	34.25%	48.20%	0.4004	15.16%	21.28%	0.1770	11.88%	17.33%	0.1410	1.10%	1.49%	0.0127
	KLP	34.75%	48.84%	0.4061	15.76%	22.15%	0.1842	8.12%	11.41%	0.0949	4.19%	5.88%	0.0489
	KLIP	35.10%	49.34%	0.4102	16.36%	23.00%	0.1912	8.27%	11.62%	0.0966	3.08%	4.33%	0.0360
	C-value	29.02%	40.97%	0.3398	16.31%	24.06%	0.1944	10.45%	18.06%	0.1324	5.17%	5.21%	0.0519
	NC-value	30.29%	42.58%	0.3540	19.20%	26.98%	0.2243	9.54%	13.40%	0.1114	3.79%	5.33%	0.0443
	HDCI	29.54%	41.53%	0.3453	17.91%	25.17%	0.2092	10.31%	14.49%	0.1205	5.06%	7.11%	0.0591
	KLIP-HD	40.54%	56.98%	0.4737	15.43%	21.69%	0.1803	5.85%	8.22%	0.0684	1.00%	1.40%	0.0117
After term border recognition	FP	36.53%	34.80%	0.3564	21.89%	21.25%	0.2157	16.39%	15.28%	0.1582	12.97%	11.03%	0.1192
	CB	25.71%	23.82%	0.2473	23.03%	21.33%	0.2215	21.01%	19.46%	0.2021	19.15%	17.73%	0.1841
	KLI	41.63%	38.57%	0.4004	21.43%	21.69%	0.2156	15.51%	20.90%	0.1781	2.86%	1.20%	0.0169
	KLP	46.78%	43.34%	0.4499	23.10%	21.39%	0.2221	11.69%	10.83%	0.1124	7.33%	6.79%	0.0705
	KLIP	45.42%	42.08%	0.4369	24.04%	22.27%	0.2312	12.80%	11.85%	0.1231	6.64%	6.15%	0.0638
	C-value	37.09%	34.39%	0.3569	23.04%	24.20%	0.2361	19.11%	15.89%	0.1735	8.78%	7.87%	0.0830
	NC-value	35.25%	32.66%	0.3391	29.10%	26.95%	0.2799	16.52%	15.31%	0.1589	8.03%	7.43%	0.0772
	HDCI	38.57%	35.73%	0.3710	26.86%	24.88%	0.2583	15.96%	14.78%	0.1534	7.52%	6.97%	0.0723
	KLIP-HD	51.23%	47.47%	0.4928	23.44%	21.71%	0.2255	10.68%	9.89%	0.1027	3.54%	3.28%	0.0340

The results shown in **Table 2** verify the effectiveness of the term border extraction we propose in this paper. The performance of each term extraction method is improved by applying term border extraction.

4.3. Pattern Construction for Candidate Extraction and Evaluation

In this section, we construct the POS patterns for candidate term extraction and evaluate the effect of the POS patterns we propose in this paper.

4.3.1 Datasets for pattern construction

To carry out the pattern construction process, we extract 37296 keywords from Chinese scientific articles (the keywords come from 17 academic journals, and these keywords cover many of the domains, including management science, computer science, economics, information science, statistics and library science).

4.3.2 Pattern construction for single-word terms and multiword terms

We first define the length of the terms we need to extract. Therefore, we conduct a statistical analysis of the word length of the terms in the keyword set we collect. The results are shown in **Table 3**.

Table 3. Number of terms classified by length.

Terms length	Number of terms
1-word	1219
2,3,4,5,6-words	36067
7-words	10

The results indicate that most Chinese terms consist of 1-6 words. Therefore, we construct POS patterns for terms less than 6 words. We discuss single-word terms and multiword terms (lengths from 2-6) separately as two different categories.

For the single-word term, we compute the number of POS tags. **Table 4** shows the patterns of the single-word terms on the list. The patterns among the 5 highest frequencies are selected to build the single-word term patterns.

Table 4. Patterns of the 1-word term on the keyword list.

1-word POS tags	Number	Probability
n (noun)	1154	1154/1219=0.9467
nl (noun locution)	4	4/1219=0.0033
v (verb)	33	33/1219=0.0271
vi (intransitive verb), vn (noun-verb)	21	21/1219=0.0172
others	7	7/1219=0.0057

Notes: When a noun location belongs to a noun, it is a special kind of noun tagged by POS tag tools. Meanwhile, the intransitive verb and noun verb belong to the verb.

Multiple words consist of two or more words. According to the position of each word that appears in the terms, we define the words in the terms as first words, last words and intermediate words (words in the middle of the terms that consist of more than 3 words). To build the multiword term patterns, we count the number of POS tags of words (first word, last word and intermediate words) in terms (**Table 5**) and then calculate the patterns of each 2-gram contained in the term (**Table 6**) to find the pattern of each 2-gram phrase inside the terms.

Table 5. POS tags of words inside the terms.

	POS tags	Number
First word	noun (includes all subtypes)	21620
	verb (except Chinese verb shi and dummy verb)	8928
	adjective (except adjective locution)	1637
	distinguishing words	1394
	adverb	669
	numeral	817
	quantifier	151
	others	861
intermediate words	noun (only includes subtype nominal morpheme and proper noun)	8519
	verb (except directional verb, verb locution, Chinese verb shi, you and dummy verb)	5894
	adjective (except adjective locution)	947
	distinguishing words	383
	adverb	362
	quantifier	737
	others	960
Last word	noun (except person name, organization name)	22395
	verb (except directional verb, verb locution, adverbial verb, Chinese verb shi, you and dummy verb)	12358
	adjective (except adjective locution)	425
	quantifier	477
	distinguishing words, adverb, numeral	142
	others	280

Notes: (shi is a Chinese verb "是" (is), you is a Chinese verb "有" (is)).

Table 6. Example of 2-gram patterns.

POS patterns	number
noun + noun	14241
noun + nominal verbs	5680
nominal verbs + noun	4325
noun + verb	4164
verb + noun	4086
verb + verb	1297
distinguishing words + noun	1178
adjective + noun	1103
nominal verbs + nominal verbs	909

4.3.3. Evaluation of the linguistic filter for candidate extraction

Here, we investigate the effect of the linguistic filtering method by applying patterns for candidate term extraction. We compare the candidate extraction results obtained by using a linguistic filter with those obtained by using n-grams. By using the n-gram method, we extract both single-word terms and multiword terms, which have a maximum length of up to 6. To avoid too many useless terms in the output list, the result is based on filtering out the term that appears only in one document or appears only once in the corpus.

Table 7. Results of the linguistic filtering method.

		Count	Precision	Recall	F1
Before removing terms appears only once or in one document	N-gram	2306030	0.14%	96.96%	0.0029
	POS patterns	456306	0.71%	94.91%	0.0141
After removing terms appears only once or in one document	N-gram	70423	4.49%	92.63%	0.0857
	POS patterns	23674	13.08%	90.64%	0.2286

Table 7 shows the precision, recall and F1 comparison for the candidate term extracted with N-grams and POS patterns. We can see that applying POS patterns can improve the precision of candidate term extraction. By applying POS patterns, we obtain a much smaller output set, which will shorten the processing time.

4.4. Evaluation of the Special Candidate Term Filtering Method

In this subsection, we evaluate the effect of filtering out some special candidate terms. As described above, special candidate terms include low-frequency terms and substrings that are contained in other candidate terms. The evaluation is based on the candidate term set generated after filtering methods of POS pattern filtering and term border recognition.

We filter out some low-frequency terms (candidate terms that only appear in one document or whose number of times in the corpus is only once). The ratio of the parent term to the substring term is set between [0.6, 0.9].

Table 8. Evaluation of special candidate term filtering methods.

		Precision	Recall	F1
Before filtering		1.43%	87.71%	0.0282
Filtering out low frequency candidate terms (top-20%, DF <5, TF <0.85)		18.32%	83.82%	0.3006
Filtering out substring terms	ratio=0.6	22.16%	80.30%	0.3474
	ratio=0.7	**22.23%**	**82.35%**	**0.3500**
	ratio=0.75	22.05%	82.67%	0.3481
	ratio=0.8	21.81%	82.94%	0.3454
	ratio=0.9	21.37%	83.17%	0.3401

Table 8 shows the results of filtering out the special candidate terms. The precision of the candidate terms obtained by filtering out low-frequency candidate terms significantly improved. This proves that most of the low-frequency terms (those that appear only in one document or once in the corpus) are not useful terms or domain-related terms. Then, we set the ratio of parent terms to substrings to 0.7 because we can obtain better precision for the candidate term set with a reasonable decrease in the recall rate.

4.5. Evaluation of Hellinger distance for Context Information Acquisition

In this subsection, we address three research questions of context information acquisition with a series of experiments and then evaluate the effect of the context information acquisition proposed in this paper.

4.5.1 Will the performance of term extraction be improved by using the context words of “important” terms?

To generate a standard context word list, two methods have been described previously. The first way is to extract the important terms that are more likely to be domain-relevant terms among all the candidates.

First, we need to determine the “important” terms. We use the C value to rank the candidate terms. We then verify the effectiveness of generating the standard context word list in the first way we listed above (extracting the context words from the context of "important" terms by experiment). To obtain the "important" term list, we set the value of the ranked candidate terms by the C value to {top-75%, top-50%, top-25%, top-10%}. The context words can be collected from the words before or after the terms. Therefore, we evaluate the term extraction based on the context word set before the terms, after the terms and the combination of both sets. The results are computed by the Hellinger distance for context information acquisition.

Table 9. Term scoring by context information using the context words of "important" terms.

		Context words before the term			Context words after the term			Combination		
		Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
All the context words of candidate terms	1	36.02%	33.36%	0.3464	32.13%	29.76%	0.3090	38.57%	35.73%	0.3710
	2	26.10%	24.17%	0.2510	27.11%	25.11%	0.2607	26.86%	24.88%	0.2583
	3	17.03%	15.77%	0.1638	19.87%	18.41%	0.1911	15.96%	14.78%	0.1534
	4	9.76%	9.04%	0.0939	9.79%	9.07%	0.0942	7.52%	6.97%	0.0723
top-75%	1	36.40%	33.39%	0.3483	32.54%	30.14%	0.3130	38.85%	36.00%	0.3737
	2	25.98%	24.29%	0.2511	26.95%	24.96%	0.2592	26.89%	24.90%	0.2586
	3	17.03%	15.77%	0.1638	19.87%	18.41%	0.1911	15.80%	14.63%	0.1519
	4	9.60%	8.90%	0.0924	9.54%	8.84%	0.0918	7.36%	6.82%	0.0708
top-50%	**1**	**36.39%**	**33.48%**	**0.3487**	**33.91%**	**29.50%**	**0.3155**	38.79%	35.94%	0.3731
	2	25.89%	24.14%	0.2499	25.89%	25.43%	0.2566	26.86%	24.88%	0.2583
	3	17.13%	15.86%	0.1647	20.09%	18.61%	0.1933	15.92%	14.75%	0.1531
	4	9.57%	8.87%	0.0920	9.51%	8.81%	0.0914	7.33%	6.79%	0.0705
top-25%	1	36.09%	33.30%	0.3464	34.03%	30.17%	0.3198	38.76%	35.91%	0.3728
	2	26.16%	24.32%	0.2520	25.64%	24.76%	0.2519	26.95%	24.96%	0.2592
	3	17.16%	15.89%	0.1650	20.19%	18.70%	0.1942	15.96%	14.78%	0.1534
	4	9.54%	8.84%	0.0918	9.41%	8.72%	0.0905	7.24%	6.70%	0.0696
top-10%	1	36.22%	32.78%	0.3441	34.06%	31.20%	0.3256	**38.91%**	**36.06%**	**0.3743**
	2	26.34%	24.96%	0.2563	25.44%	23.82%	0.2460	26.92%	24.93%	0.2589
	3	16.94%	15.69%	0.1629	20.06%	18.58%	0.1930	15.80%	14.63%	0.1519
	4	9.63%	8.93%	0.0927	9.44%	8.75%	0.0908	7.27%	6.73%	0.0699

Table 9 presents and compares the results by defining the threshold of "important" terms from which the context word list is extracted. We compare the results generated by context information with or without the use of "important" terms. The results show that the best performance of term extraction is achieved by using the context word list extracted from the top 75% of the terms ranked by the C value. However, it only slightly improves in comparison with the results achieved by extracting the context words from all the candidate terms. Therefore, we do not need to extract the context word list from the "important" terms.

4.5.2 Will it improve the performance of term extraction by filtering out some low-frequency words as the context words list?

The second way to generate a standard context word list we describe above is to filter out the low-frequency words in the selected context words (context words of important terms or all the candidate terms). Here, we expect to determine whether filtering out low-frequency words as a context word list improves the performance of term extraction via context information.

Table 10. Term scoring by context information using the context word list by filtering out low-frequency words.

		Context words before the term			Context words after the term			Combination		
		Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Before filtering out low frequency words from the standard context word list	1	36.02%	33.36%	0.3464	**32.13%**	**29.76%**	**0.3090**	**38.57%**	**35.73%**	**0.3710**
	2	26.10%	24.17%	0.2510	27.11%	25.11%	0.2607	26.86%	24.88%	0.2583
	3	17.03%	15.77%	0.1638	19.87%	18.41%	0.1911	15.96%	14.78%	0.1534
	4	9.76%	9.04%	0.0939	9.79%	9.07%	0.0942	7.52%	6.97%	0.0723
filter out low frequency words tf <2 from the standard context word list	1	**36.05%**	**33.39%**	**0.3467**	32.04%	29.68%	0.3081	38.57%	35.73%	0.3710
	2	25.94%	24.03%	0.2495	27.05%	25.05%	0.2601	26.76%	24.79%	0.2574
	3	17.16%	15.89%	0.1650	20.03%	18.55%	0.1926	16.02%	14.84%	0.1541
	4	9.76%	9.04%	0.0939	9.79%	9.07%	0.0942	7.55%	6.99%	0.0726
filter out low frequency words tf <3 from the standard context word list	1	**36.05%**	**33.39%**	**0.3467**	32.10%	29.73%	0.3087	38.44%	35.62%	0.3697
	2	25.88%	23.97%	0.2489	26.79%	24.82%	0.2577	**27.05%**	**25.05%**	**0.2601**
	3	17.25%	15.98%	0.1659	20.19%	18.70%	0.1942	15.80%	14.63%	0.1519
	4	9.73%	9.01%	0.0936	9.82%	9.10%	0.0945	7.61%	7.05%	0.0732

Table 10presents the results of the term extraction based on context information, with or without filtering out the low-frequency words in the context word list. The results show no significant improvement after filtering out low-frequency words from the context word list. Therefore, we do not need to filter out the low-frequency words from the context word list.

4.5.3 Do we need a strict syntactic structure of context words to nouns, adjectives and verbs?

We evaluate the HDCI measure with context words that are restricted to only nouns, adjectives and verbs and compare the results with those of the HDCI with those of all context words.

Table 11. Comparison of specific syntactic structure words with all words as context words for term scoring on the basis of context information.

		Context words before the term			Context words after the term			Combination		
		Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
context information with all the words	1	36.02%	33.36%	0.3464	32.13%	29.76%	0.3090	38.57%	35.73%	0.3710
	2	26.10%	24.17%	0.2510	27.11%	25.11%	0.2607	26.86%	24.88%	0.2583
	3	17.03%	15.77%	0.1638	19.87%	18.41%	0.1911	15.96%	14.78%	0.1534
	4	9.76%	9.04%	0.0939	9.79%	9.07%	0.0942	7.52%	6.97%	0.0723
context information with only nouns, adjectives and verbs	1	25.37%	23.50%	0.2440	26.67%	24.70%	0.2565	29.66%	27.48%	0.2853
	2	26.79%	24.82%	0.2577	24.99%	23.15%	0.2404	27.74%	25.70%	0.2668
	3	22.44%	20.78%	0.2158	24.17%	22.39%	0.2325	20.16%	18.67%	0.1939
	4	14.30%	13.26%	0.1376	13.08%	12.12%	0.1258	11.34%	10.51%	0.1091

Table 11 shows the results of HDCI with context words specific to nouns, adjectives and verbs. The HDCI method, which uses context words that are not restricted to any specific syntactic structure, achieves better results. This proves that context words, not only nouns but also adjectives and verbs, provide clues for extracting terms.

4.5.4. Comparison with other method-based context information

We evaluate two methods based on context information for term extraction in Section 3.2: the NC value and the left/right branching entropy.

We compare the results of the methods above with those of our method (Hellinger distance for context information, HDCI). We conducted two experiments on term scoring-based N values: one used context words whose syntactic structure included nouns, adjectives and verbs, and the other used all the context words of the term. The LR-entropy in Table 12 refers to the left/right branching entropy method.

Table 12. Comparison of term scoring methods on the basis of context information.

	Top 25%			Top 25%-50%			Top 50%-75%			Top 75%-100%		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
HDCI	**38.57%**	**35.73%**	**0.3710**	26.86%	24.88%	0.2583	15.96%	14.78%	0.1534	7.52%	6.97%	0.0723
N-value (n, adj, v)	21.10%	19.55%	0.2029	24.08%	22.30%	0.2315	25.06%	23.21%	0.2410	18.67%	17.30%	0.1796
N-value	29.66%	27.48%	0.2853	30.93%	28.65%	0.2975	20.22%	18.73%	0.1945	8.09%	7.49%	0.0778
LR-entropy	33.65%	31.17%	0.3236	23.53%	21.60%	0.2252	16.66%	15.57%	0.1610	15.12%	14.02%	0.1455

Table 12 presents the results of the term scoring methods, which are based on context information. Among these methods, HDCI achieves the best results. This proves that the Hellinger distance for context information is effective and outperforms other methods based on context information.

4.6. valuation of term scoring methods

In this section, we evaluate the term scoring methods we propose in this article with eight other methods. We evaluate those methods for single-word term extraction and multiword term extraction.

4.6.1 Evaluation of single-word term extraction

For single-word term extraction, because KIP cannot be applied to extract single-word terms, we evaluate the other eight methods. Table 13 shows the precision, recall and F1 comparison for the single-word term extracted with 8 different term scoring methods.

Table 13. Comparison of single-word term extraction by 8 methods.

	Top 25%			Top 25%-50%			Top 50%-75%			Top 75%-100%		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
FP	16.91%	30.32%	0.2171	5.93%	10.64%	0.0762	7.74%	13.83%	0.0992	6.55%	11.70%	0.0840
CB	9.20%	16.49%	0.1181	11.57%	20.74%	0.1486	10.42%	18.62%	0.1336	5.95%	10.64%	0.0763
KLI	20.47%	36.70%	0.2629	10.39%	18.62%	0.1333	4.76%	8.51%	0.0611	1.49%	2.66%	0.0191
KLIP	20.18%	36.17%	0.2590	9.79%	17.55%	0.1257	5.36%	9.57%	0.0687	1.79%	3.19%	0.0229
C-value	14.84%	26.60%	0.1905	10.09%	18.09%	0.1295	7.80%	14.36%	0.1011	4.29%	7.45%	0.0545
NC-	12.76%	22.87%	0.1638	10.09%	18.09%	0.1295	9.82%	17.55%	0.1260	4.46%	7.98%	0.0573
HDCI	15.13%	27.13%	0.1943	10.68%	19.15%	0.1371	7.44%	13.30%	0.0954	3.87%	6.91%	0.0496
KLIP-HD	20.47%	36.70%	0.2629	9.20%	16.49%	0.1181	5.65%	10.11%	0.0725	1.79%	3.19%	0.0229

Notes: Dividing the result of the term scoring method into 4 intervals)

The best results are achieved via KLI. Both KLIP-HD and KLI achieved the best results in the top 25% interval. However, in the interval from the top 25% to the top 75%, the performance of KLI is better than that of KLIP-HD. This is because single-word term extraction is strongly affected by the informativeness factor. High-term frequency is an important feature of single-word terms, and the difference in term frequency between two corpora is another feature of single-word terms. KLI is a method that considers both the features of single-word term extraction, so it achieves better results.

We compare 8 term scoring methods for extracting the top single-word terms. Table 14 presents the results of the single-word term extraction.

Table 14. Precision comparison of 8 term scoring methods for single-word terms.

	FP	CB	KLI	KLIP	C-value	NC-value	HDCI	KLIP-HD
100	0.2475	0.0693	**0.3663**	0.3663	0.2970	0.2871	0.2871	0.3861
200	0.2139	0.0846	**0.2736**	0.2687	0.1642	0.1642	0.1940	0.2687
300	0.1761	0.0930	**0.2193**	0.2159	0.1462	0.1296	0.1595	0.2126
400	0.1496	0.0973	**0.1895**	0.1945	0.1471	0.1247	0.1421	0.1970
500	0.1317	0.0978	**0.1717**	0.1677	0.1355	0.1138	0.1437	0.1816
600	0.1215	0.0982	**0.1595**	0.1581	0.1298	0.1148	0.1348	0.1631
700	0.1113	0.1084	**0.1496**	0.1455	0.1252	0.1170	0.1255	0.1427
800	0.1124	0.1099	**0.1409**	0.1361	0.1177	0.1161	0.1199	0.1336
900	0.1065	0.1043	**0.1299**	0.1265	0.1110	0.1165	0.1154	0.1276
1000	0.1019	0.1039	**0.1199**	0.1189	0.1091	0.1079	0.1119	0.1189

As shown in Table 14, we can see that KLI achieved the best results. These precision results are also shown in Figure 1 for the single-word terms.

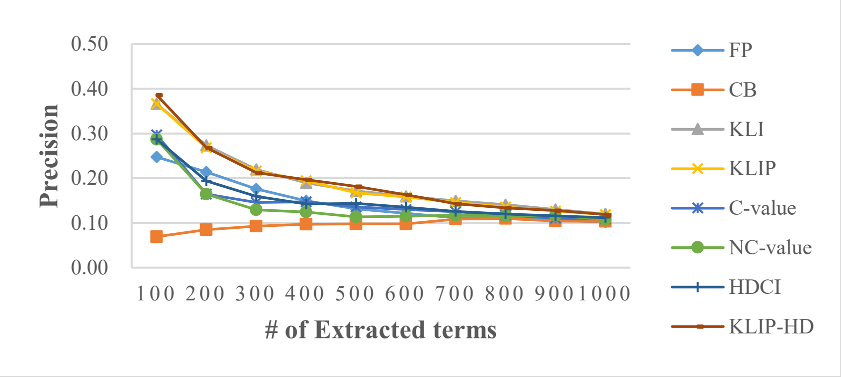

Figure 1. Precision comparison of 8 term scoring methods for single-word terms.

4.6.2 Evaluation of multiword term extraction

For multiword term extraction, we evaluate nine methods. Table 15 presents the term ranking results in comparison with those of the 9 term scoring methods.

Table 15. Comparison of 9 term scoring methods for extracting multiword terms

	Top 25%			Top 25%-50%			Top 50%-75%			Top 75%-100%		
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
FP	37.24%	32.98%	0.3498	22.21%	20.75%	0.2146	16.75%	14.09%	0.1531	17.39%	15.21%	0.1623
CB	42.33%	37.41%	0.3972	19.20%	16.97%	0.1802	17.14%	16.17%	0.1664	15.14%	12.48%	0.1368
KLI	47.27%	42.30%	0.4465	20.16%	17.71%	0.1886	15.81%	21.80%	0.1833	3.16%	1.21%	0.0175
KLP	47.76%	42.21%	0.4481	24.98%	22.08%	0.2344	13.10%	11.58%	0.1230	8.10%	7.15%	0.0760
KLIP	48.39%	42.77%	0.4541	24.81%	21.93%	0.2328	13.56%	11.99%	0.1272	7.19%	6.35%	0.0674
C-value	44.90%	40.79%	0.4275	22.65%	19.51%	0.2096	15.30%	17.68%	0.1641	8.28%	5.05%	0.0627
NC-value	45.83%	40.51%	0.4301	25.54%	22.58%	0.2397	15.42%	13.63%	0.1447	7.15%	6.32%	0.0671
HDCI	45.52%	40.23%	0.4271	26.24%	23.20%	0.2463	15.10%	13.35%	0.1417	7.08%	6.26%	0.0664
KLIP-HD	53.96%	47.69%	0.5063	24.98%	22.08%	0.2344	11.21%	9.91%	0.1052	3.79%	3.34%	0.0355

Notes: Dividing the results of the term scoring method into 4 intervals.

As shown in Table 15, KLIP-HD achieves the best results. The precision of KLIP-HD is 5.57% higher than the second-highest results achieved by KLIP, and the recall of KLIP-HD is 5.48% higher than that of KLIP. They achieved the same results in the interval of the top 25% to the top 50%. We can see that KLIP-HD is more effective in multiterm extraction than the other methods are. KLIP-HD considers both informativeness and phraseness, so it performs better than other methods do. KLIP also considers both informativeness and phraseness, but it does not consider context information. The results prove that our method is better than other methods in terms of term scoring.

We compare 9 term scoring methods for extracting the top multiword terms. Table 16 shows the precision comparison for the multiword term extraction.

Table 16. Precision comparison of 9 term scoring methods for multiword terms

	FP	CB	KLI	KLP	KLIP	C-value	NC-value	HDCI	KLIP-HD
100	0.2871	0.4554	0.7525	0.6634	0.6931	0.7228	0.6733	**0.8515**	0.7228
200	0.3333	0.5373	0.7463	0.6766	0.7065	0.6766	0.6418	0.7711	0.7363
300	0.3953	0.5615	0.7467	0.6910	0.7043	0.6502	0.6047	0.7375	0.7276
400	0.4190	0.5661	0.7239	0.6559	0.6883	0.6593	0.6284	0.7207	**0.7307**
500	0.4411	0.5808	0.7134	0.6507	0.6766	0.6567	0.6048	0.6986	**0.7206**
600	0.4293	0.5824	0.7072	0.6389	0.6855	0.6528	0.6023	0.6839	**0.7288**
700	0.4223	0.5678	0.6886	0.6362	0.6619	0.6430	0.6049	0.6591	**0.7218**
800	0.4239	0.5680	0.6833	0.6367	0.6604	0.6392	0.5955	0.6367	**0.7154**
900	0.4279	0.5638	0.6759	0.6249	0.6504	0.6268	0.5949	0.6182	**0.6970**
1000	0.4168	0.5654	0.6581	0.6184	0.6444	0.6233	0.5924	0.6034	**0.6803**
2000	0.4113	0.4858	0.5401	0.5282	0.5492	0.5144	0.5082	0.5102	**0.6002**
3000	0.3645	0.4167	0.4535	0.4672	0.4728	0.4469	0.4499	0.4502	**0.5298**
4000	0.3330	0.3582	0.3919	0.4254	0.4256	0.3946	0.4104	0.4126	**0.4686**
5000	0.3095	0.3305	0.3636	0.3853	0.3897	0.3589	0.3783	0.3825	**0.4257**
6000	0.2923	0.3002	0.3141	0.3553	0.3569	0.3212	0.3501	0.3508	**0.3834**

As shown in Table 16, we can see that HDCI achieved the best results for the top-100, top-200 and top-300 intervals. KLIP-HD obtains the best results for the other intervals for multiword term scoring. These precision results are also shown in Figure 2 for the multiword terms.

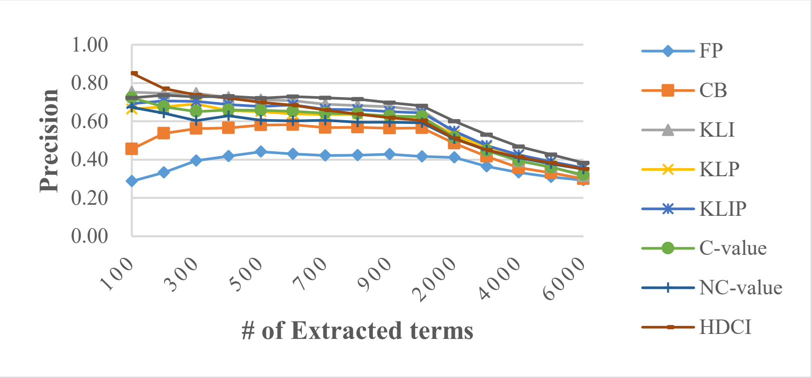

Figure 2. Precision comparison of 9 term scoring methods for multiword terms.

Table 17 lists the top-20 ranked multiword terms extracted via the 9 methods. The irrelevant terms are italicized and marked with *. We can see that KLIP-HD prefers to extract longer terms than the other methods do.

Table 17. Top-20 terms of the candidate terms ranked by 9 methods for multiword term extraction

KLI	KLP	KLIP	C-value	NC-value	HDCI	KLP-HD	FP	CB
情报学报*	情报学报*	情报学报*	情报学报*	情报学报*	聚类（cluster)	关联规则挖掘（association rules mining)	发展（development)*	本文（this paper) *
相似度（similarity)	相似度（similarity)	相似度（similarity)	相似度（similarity)	相似度（similarity)	链接（link)	聚类（cluster)	包括（include)*	所示（it shows)*
聚类（cluster)	链接（link)	聚类（cluster)	聚类（cluster)	聚类（cluster)	知识网络（knowledge network)	科技报告（scientific and technical report)	主义（-ism)*	本文提出（this paper proposes)*
科技报告（scientific and technical report)	聚类（cluster)	科技报告（scientific and technical report)	链接（link)	链接（link)	关键词（keywords)	可视化（visualization)	问题（problems)*	发展（development)*
万方数据*	微博（Microblog)	链接（link)	科技报告（scientific and technical report)	科技报告（scientific and technical report)	知识组织（knowledge organization)	竞争情报（competitive intelligence)	例如（for example)*	相似度（similarity)
信息资源(information resource)	科技报告（scientific and technical report)	微博（Microblog)	微博（Microblog)	微博（Microblog)	信息资源检索（information resources retrieval)	相似度（similarity)	发生（occur)*	问题（problems)*
微博（Microblog)	可视化（visualization)	万方数据*	万方数据*	万方数据*	相似度（similarity)	作者学术影响力双重测度探讨（measurement in academic influence of the author)*	决定（decision)*	描述（description)*
链接（link)	万方数据*	信息资源（information resources)	信息资源（information resources)	信息资源（information resources)	数据挖掘（data mining)	共词网络（coword network)	要求（requirements)*	包括（include)*
共词（coword)	主要研究方向（research interests)	可视化（visualization)	可视化（visualization)	卜文期干刊*	竞争情报（competitive intelligence)	共词分析（coword analysis)	如果（if)*	链接（link)
竞争情报（competitive intelligence)	收稿日期（receiving date of article)*	主要研究方向（research interests)*	主要研究方向（research interests)*	可视化（visualization)	文本挖掘（data mining)	聚类分析(cluster analysis)	导致（result in)*	科技报告（scientific and technical report)
知识网络（knowledge network)	信息资源（information resources)	突发事件（emergency)	共词（coword)	万方数据情报学报*	知识管理（knowledge management)	知识网络（knowledge network)	需要（need)*	聚类（cluster)
信息资源检索（information resources retrieval)	突发事件（emergency)	收稿日期（receiving date of article)*	参考文献（references)	卜文期干干*	科技报告（scientific and technical report)	科研团队动态演化规律研究（research of dynamic evolution of scientific research team)*	发表（publish)*	该方法（this method)*
参考文献（references)*	权重（weight)	共词（coword)	信息资源检索（information resources retrieval)	卜文期刊*	可视化（visualization)	链接分析（link analysis)	文件（documents)*	微博（Microblog)*

Table 18. Top-20 terms of the candidate terms ranked by 9 methods for multiword term extraction (Continued)

KLI	KLP	KLIP	C-value	NC-value	HDCI	KLP-HD	FP	CB
突发事件（emergency)	参考文献（references)*	竞争情报（competitive intelligence)	权重（weight)	责任编辑（editor in charge)*	复杂网络（complex network)	知识组织（knowledge organization)	类似（similar)*	发表（publish)*
数字图书馆（digital library)	研究*	权重（weight)	突发事件（emergency)	主要研究方向（research interests)*	引文网络（citation network)	虚拟社区知识共享水平 (knowledge sharing level of virtual community)	实施（implement)*	发生（occur)*
知识管理（knowledge management)	链接分析（link analysis)	参考文献（references)*	竞争情报（competitive intelligence)	共词（coword)	微博（Microblog)	评价方法（evaluation methodology)	描述（description)*	如果（if)*
图书情报（library and information)	竞争情报（competitive intelligence)	研究*	知识网络（knowledge network)	参考文献（references)*	共词网络（coword network)	潜在主题可视化（visualization)	讨论（discussion)*	万方数据*
研究 *	图书情报（library and information)	数字图书馆（digital library)	信息资源（information resources)服务	信息资源（information resources)检索	知识发现（knowledge discovery)	知识管理（knowledge management)	高度（height)*	度计算(compute)*
社会网络	数字图书馆（digital library)	图书情报（library and information)	维度（dimensionality)	权重（weight)	共词（coword)分析	数字图书馆（digital library)	性质（characteristic)*	例如（for example)*
主要研究方向（research interests)	科研机构（scientific institution)	信息资源（information resources)检索	图书情报（library and information)	文献链接作者(author of literature links)*	特征词（feature word)	收稿日期（receiving date of article)*	情报学报*	信息资源（information resources)
权重（weight)	共词（coword)	链接分析(link analysis)	信息资源管理 (information resources management)	信息资源管理 (information resources management)	聚类（cluster)分析	信息资源检索（information resources retrieval)	逐步（step by step)*	导致（result in)*

Notes: “卜文期干刊 Chinese garbage characters” is a Chinese scientific journal; “万方数据（Wanfang data)” is a Chinese database; “情报学报（Journal of the China Society for Scientific and Technical Information)” is a Chinese scientific journal.

5. Conclusion

In this paper, we propose a term border recognition method to define the border of the candidate terms. By recognizing domain-independent high-frequency words and low-frequency words as term border words, we can split the sentences in the domain corpus into smaller segments, avoid the extraction of some useless candidate terms and shorten the processing time of the next phrases. We define the thresholds of high-frequency words and low-frequency words as term border words and then evaluate the effect of term border recognition by comparing the effects of term extraction before and after term border recognition. The results show that term border recognition can improve the performance of term extraction.

After term border recognition, we design POS patterns for Chinese candidate term extraction. We compare the candidate term extraction via POS patterns and n-gram methods. The results indicate that POS patterns can filter out many useless terms with very little loss of useful terms. Therefore, it can greatly improve the precision of term extraction.

In terms of the scoring phase, we propose a new term ranking method called context information acquisition to make use of the context information of candidate terms. We use the Hellinger distance to measure the difference between the context word list of each candidate term and the context word list of all the candidate terms. We investigate the influence of the factors on context information acquisition: context words of “important” terms as standard context words list, high-frequency words as standard context words list, and POS tags of the context words. The results indicate that neither selecting context words of “important” terms nor removing low-frequency words as standard context words improve the performance of the term extraction, and we do not need to restrict the syntactic structure of context words to nouns, adjectives and verbs because it does not help at all. Finally, we compare our context information acquisition method with other methods that are based on context information of terms such as the N value and LR-entrophy. The results show that our method outperforms other methods on the basis of context information.

To combine the informativeness and phraseness of terms, we combine Kullback–Leibler divergence for scoring both informativeness and phraseness and context information acquisition-based Hellinger distance for term scoring. We evaluate the term scoring method we proposed with 8 other methods for single-word term extraction and multiword term extraction. The results indicate that our method outperforms other methods, especially in terms of multiterm extraction.

Acknowledgments: This work was supported by the Department of Education Funded Project for Philosophy and Social Sciences in Colleges and Universities of Jiangsu Province, Grant 2020SJA0288.

References

1. Bourigault, D. Surface grammatical analysis for the extraction of terminological noun phrases. In Proceedings of the 14th International Conference on Computational Linguistics. Nantes, France, 1992, 977-981. https://doi.org/10.3115/993079.993111
2. Justeson, J. S., Katz, S. M. Technical terminology: some linguistic properties and an algorithm for identification in text. Natural language engineering, 1995, 1(1), 9-27. https://doi.org/10.1017/S1351324900000048
3. Verberne, S., Sappelli, M., Hiemstra, D., et al. Evaluation and analysis of term scoring methods for term extraction. Information Retrieval Journal, 2016, 19(5), 510-545. https://doi.org/10.1007/s10791-016-9286-2
4. Salton, G., Buckley, C. Term-weighting approaches in automatic text retrieval. Information processing & management, 1988, 24(5): 513-523. https://doi.org/10.1016/0306-4573(88)90021-0
5. Frantzi, K., Ananiadou, S., Mima, H. Automatic recognition of multi-word terms: The C-value/NC-value method. International journal on digital libraries, 2000, 3(2), 115-130. https://doi.org/10.1007/s007999900023
6. Church, K., Hanks, P. (1990). Word association norms, mutual information, and lexicography. Computational linguistics, 1990, 16(1), 22-29.

7. Pantel, P., Lin, D. A statistical corpus-based term extractor. In Proceedings of the14th biennial conference of the Canadian society on computational studies of intelligence: Advances in artificial intelligence. Ottawa, 2001, 36-46. https://doi.org/10.1007/3-540-45153-6_4
8. Matsuo, Y., Ishizuka, M. Keyword extraction from a single document using word co-occurrence statistical information. International Journal on Artificial Intelligence Tools, 2004, 13(01), 157-169. https://doi.org/10.1142/S0218213004001466
9. Tomokiyo, T., Hurst, M. A language model approach to keyphrase extraction. In Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment, 2003, 33-40. https://doi.org/10.3115/1119282.1119287
10. Kovačević, A., Konjović, Z., Milosavljević, B., et al. Mining methodologies from NLP publications: A case study in automatic terminology recognition. Computer Speech & Language, 2012, 26(2), 105-126. https://doi.org/10.1016/j.csl.2011.09.001
11. Judea, A., Schütze, H., Brügmann, S. Unsupervised training set generation for automatic acquisition of technical terminology in patents. In Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical Papers, 2014: 290-300.
12. Conrado, M., Pardo, T., Rezende, S. O. A machine learning approach to automatic term extraction using a rich feature set. In Proceedings of the 2013 NAACL HLT student research workshop, 2013, 16-23.
13. Lossio-Ventura, J. A., Jonquet, C., Roche, M., et al. Yet another ranking function for automatic multiword term extraction. In International conference on natural language processing. Cham, 2013, 52-64. https://doi.org/10.1007/978-3-319-10888-9_6
14. Lossio-Ventura, J. A., Jonquet, C., Roche, M., et al. Biomedical term extraction: overview and a new methodology. Information Retrieval Journal, 2016, 19(1), 59-99. https://doi.org/10.1007/s10791-015-9262-2
15. Ittoo, A., Bouma, G. Term extraction from sparse, ungrammatical domain-specific documents. Expert Systems with Applications, 2013, 40(7), 2530-2540. https://doi.org/10.1016/j.eswa.2012.10.067
16. Bolshakova, E., Loukachevitch, N., Nokel, M. Topic models can improve domain term extraction. In European Conference on Information Retrieval. Berlin, Heidelberg, 2013, 684-687. https://doi.org/10.1007/978-3-642-36973-5_60
17. Turney, P. D. Learning algorithms for keyphrase extraction. Information retrieval, 2000, 2(4), 303-336. https://doi.org/10.1023/A:1009976227802
18. Wermter, J., Hahn, U. You can't beat frequency (unless you use linguistic knowledge)–a qualitative evaluation of association measures for collocation and term extraction. In Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, 2003: 785-792. https://doi.org/10.3115/1220175.1220274
19. Yang, Y., Lu, Q., Zhao, T. A delimiter-based general approach for Chinese term extraction. Journal of the American society for information science and technology, 2010, 61(1), 111-125. https://doi.org/10.1002/asi.21221
20. Zhou, L., Zhang, D. NLPIR: A theoretical framework for applying natural language processing to information retrieval. Journal of the American Society for Information Science and Technology, 2003, 54(2), 115-123. https://doi.org/10.1002/asi.10193
21. Hellinger, E. Neue begründung der theorie quadratischer formen von unendlichvielen veränderlichen. Journal für die reine und angewandte Mathematik, 1909, 1909(136), 210-271. https://doi.org/10.1515/crll.1909.136.210
22. Csiszár, I., Shields, P. C. Information theory and statistics: A tutorial. Foundations and Trends® in Communications and Information Theory, 2004, 1(4), 417-528. https://doi.org/10.1561/0100000004
23. Chen, Y. N., Huang, Y., Kong, S. Y., et al. Automatic key term extraction from spoken course lectures using branching entropy and prosodic/semantic features. In 2010 IEEE Spoken Language Technology Workshop. IEEE, 2010, 265-270. https://doi.org/10.1109/SLT.2010.5700862
24. Zhang, Z., Iria, J., Brewster, C., et al. A comparative evaluation of term recognition algorithms. In LREC (Vol. 5), 2008.
25. Rayson, P., Garside, R. Comparing corpora using frequency profiling. In Proceedings of the Workshop on Comparing Corpora, held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics (ACL 2000), 1-8 October 2000, Hong Kong. https://doi.org/10.3115/1117729.1117730
26. Matsuo, Y., Ishizuka, M. Keyword extraction from a single document using word co-occurrence statistical information. International Journal on Artificial Intelligence Tools, 2004, 13(01), 157-169. https://doi.org/10.1142/S0218213004001466

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of the BSP and/or the editor(s). BSP and/or the editor(s) disclose responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.