Chinese Text Similarity Computation Based on Sequence Alignment Algorithms (Postprint)

Authors: Zhao Dengpeng, Xiong Huixiang, Tian Fengshou, Li Xinran

Date: 2023-04-01T00:00:00+00:00

Abstract

[Purpose/Significance] In view of the application of sequence alignment algorithms in text similarity, this paper improves the global alignment algorithm to enhance its accuracy, while simultaneously applying the local alignment algorithm to effectively address the alignment of two texts with substantial content differences or length disparities.

[Method/Process] First, the CRF model in HanLP is employed to normalize the Chinese text dataset from online academic resources, constituting a Chinese sequence set. Subsequently, the latest Chinese Wikipedia corpus is utilized to train a Word2Vec model for constructing a word pair scoring matrix. Finally, based on this scoring matrix and improved scoring rules, two Chinese sequences undergoing global or local alignment are aligned to obtain the optimal solution; this optimal solution is then backtracked to retrieve its alignment path, thereby calculating the similarity between the two Chinese sequences.

[Results/Conclusion] Experimental results demonstrate that, compared with existing research on global alignment algorithms, the word pair scoring matrix constructed based on part-of-speech tagging results and Word2Vec in this paper further improves the accuracy of the global alignment algorithm for text similarity calculation. Furthermore, the local alignment algorithm applied to text similarity calculation can effectively solve the problem of aligning two texts with significant content differences or length disparities.

Full Text

Preamble

Research on Chinese Text Similarity Calculation Based on Sequence Alignment Algorithms

Zhao Dengpeng[1]*, Xiong Huixiang*[1]*, Tian Fengshou*[2]*, Li Xinran*[1]
[1]School of Information Management, Central China Normal University, Wuhan

430079
[2]Technology R&D Department, Shandong Gaoxun Zhenyuan Education Technology Co., Ltd., Jinan 250000

Abstract: [Purpose/Significance] This study aims to improve the global alignment algorithm and enhance its accuracy in text similarity applications, while applying local alignment algorithms to effectively address the comparison of texts with significant content or length differences. [Method/Process] First, the CRF model in HanLP was used to normalize the Chinese text dataset of online academic resources, constructing a Chinese sequence set. Next, the latest Chinese Wikipedia corpus was utilized to train a Word2Vec model for building a word-pair scoring matrix. Finally, based on this scoring matrix and improved scoring rules, two Chinese sequences were aligned using global/local alignment to obtain the optimal alignment solution. This optimal solution was backtracked to retrieve the alignment path, enabling calculation of the similarity between the two Chinese sequences. [Result/Conclusion] Experimental results demonstrate that compared with existing research on global alignment algorithms, our approach—incorporating part-of-speech tagging results and a Word2Vec-constructed word-pair scoring matrix—further improves the accuracy of text similarity calculation. Moreover, the local alignment algorithm applied to text similarity calculation can effectively solve problems in comparing texts with substantial content or length differences.

Keywords: CRF model; part-of-speech tagging; Word2Vec; sequence alignment; local alignment; text similarity
Classification Number: TP391.1
DOI: 10.13266/j.issn.0252-3116.2021.11.011

Introduction

With the rapid development of information technology, mining and researching the massive textual information generated on the Internet can provide users with valuable content, including text classification and clustering, personalized recommendation, information extraction, information retrieval, and search engines. As a method for measuring differences and commonalities between texts, text similarity constitutes a core component of these technical tasks [1-2]. In recent years, text similarity has been primarily applied in word sense disambiguation, automatic abstract extraction, automatic evaluation of machine translation, database schema matching, and semantic heterogeneity problems [3]. In the field of Chinese information processing, calculating similarity between Chinese strings such as words and phrases plays a crucial role in dictionary compilation, example-based machine translation, automatic question answering, and information filtering [4]. Current text similarity calculation methods mainly include string-based approaches, corpus-based approaches, knowledge base-based approaches, and hybrid methods [5-10]. Sequence alignment algorithms belong to string-based methods and demonstrate good performance for temporal and streaming data [11].

Sequence alignment algorithms originate from bioinformatics, where they represent the most common and classic research method for analyzing gene structure and function by comparing amino acid or nucleic acid sequences. This approach identifies similar regions and conserved sites between two sequences to seek homologous structures and reveal issues in biological evolution, heredity, and variation [12]. In 1970, S.B. Needleman and C.D. Wunsch proposed the dual-sequence global alignment algorithm [13]. In 1975, T.F. Smith and M.S. Waterman improved upon this work by introducing a dual-sequence local alignment algorithm [14]. Subsequently, with the continuous development of bioinformatics, numerous sequence alignment tools and software emerged in 2019 [15-19] and have been continuously refined. Recent research on sequence alignment algorithms has focused primarily on algorithmic improvements and acceleration [20-21]. In 2020, R.J. Lu and X. Zhao et al. used sequence alignment algorithms to study genetic similarities among COVID-19, SARS, and MERS viruses [22].

In the library and information science domain, Xu Shuo et al. [23] first proposed using global alignment algorithms to calculate Chinese text similarity in 2010, addressing the limitation of traditional semantic similarity methods that ignored word order. In 2014, Wang Ting et al. [24] improved the rationality and accuracy of global alignment algorithms for word comparison by referencing Tong Jiule et al.' s [25] research on synonym forests. However, constrained by the coverage and breadth of synonym forests, this method only proved effective in specific domains. Xiong Huixiang et al. [26] constructed a word-pair scoring matrix based on Word2Vec, significantly improving algorithmic accuracy and effectively handling the problem of "duplicate word pairs" in Chinese texts. Nevertheless, existing research on sequence alignment algorithms has a limitation: global alignment algorithms only work effectively when the two segmented Chinese texts have minimal content and length differences. To address this, we propose an improved local alignment algorithm for comparing texts with substantial content or length differences. To better apply sequence alignment algorithms to Chinese text similarity research, this study further enhances global alignment algorithm accuracy based on CRF model part-of-speech tagging results and a Word2Vec-constructed word-pair scoring matrix, while applying local alignment algorithms to effectively solve comparison problems between texts with significant differences, thereby improving the effectiveness and accuracy of sequence alignment algorithms for Chinese text similarity calculation.

CRF Model, Word2Vec, and Sequence Alignment Algorithms

2.1 CRF Model

With researchers continuously proposing various digital models for language information processing, statistical-based segmentation technology has become mainstream. HMM, MEMM, and CRF are three commonly used statistical mod-

els [27]. Unlike HMM, CRF does not have strict independence assumptions and can better accommodate contextual information. Simultaneously, CRF models possess the characteristics of MEMM discriminative models, requiring small data volumes while offering fast processing speed and high accuracy. Compared with traditional segmentation models and tools, CRF demonstrates certain advantages and is frequently used in natural language processing tasks such as syntactic analysis, named entity recognition, and part-of-speech tagging.

HanLP is an NLP tool developed by He Han in 2014 and open-sourced on GitHub, which includes longest matching, HMM, perceptron, CRF, and other natural language processing models. HanLP further improves the effectiveness and accuracy of HMM, CRF, and other natural language processing methods by referencing CoNLL-X [28], Biaffine [29], FastText [30], and BERT [31]. The CRF model included in HanLP provides excellent word segmentation and part-of-speech tagging capabilities and can effectively recognize out-of-vocabulary words. Therefore, we selected HanLP' s CRF model to complete a series of natural language processing tasks in our empirical study.

2.2 Word2Vec Model

Word2Vec is a word vector model developed by Google in 2013 based on deep learning concepts, primarily used to transform unstructured text information into vectorized form [32]. Since its release, Word2Vec has been widely applied in the natural language processing field, with related research gradually increasing. Word2Vec has become one of the most representative tools in natural language processing. By learning from text, Word2Vec can convert words into vector form and represent semantic information through word vectors [33]. Additionally, as a natural language processing tool, one of Word2Vec' s key characteristics is using contextual information to achieve word feature representation, thereby solving the dimensionality curse problem.

2.3 Sequence Alignment Algorithms

Sequence alignment algorithms are mainly divided into two types: global alignment algorithms that seek overall sequence similarity, and local alignment algorithms that seek local sequence similarity. Both algorithms share the same word-pair scoring matrix to compare similar words in texts, further exploring similarity relationships between texts.

2.3.1 Foundational Concepts Applying sequence alignment algorithms to Chinese text similarity research involves processing two Chinese texts into two word sequences arranged in order after segmentation, then aligning these sequences to compare their similarities. Gap symbols can be inserted into sequences to align as many identical or similar words as possible in the same column. To better illustrate how sequence alignment algorithms study Chinese text similarity, we define relevant concepts based on references [12][26]:

(1) **Chinese Sequence** cs_i ($i \in \{1, 2, 3, ..., n\}$): cs_i is a word sequence obtained from preprocessing and segmenting a Chinese text, formally represented as $cs_i = \{t_{i,1}, t_{i,2}, ..., t_{i,k}, ..., t_{i,m}\}$, where $t_{i,k}$ denotes the k-th word in cs_i, with words arranged in their original order.

(2) **Chinese Sequence Set** CS (Chinese Sequence Set): $CS = \{cs_1, cs_2, ..., cs_i, ..., cs_n\}$, where n represents the number of Chinese sequences in CS, and cs_i denotes the i-th Chinese sequence in set CS.

(3) **Alignment Matrix** M (Alignment Matrix): $M_{m \times q}$ represents the result of aligning $cs_i = \{t_{i,1}, t_{i,2}, ..., t_{i,k}, ..., t_{i,m}\}$ with $cs_j = \{t_{j,1}, t_{j,2}, ..., t_{j,p}, ..., t_{j,q}\}$. Before alignment, a row of gap symbols "-" and a column of gap symbols "-" are inserted into the matrix, making the final size of M $(m + 1) \times (q + 1)$, where $M_{k,p}$ represents the alignment between the k-th word of cs_i and the p-th word of cs_j.

(4) **Word-Pair Scoring Matrix** W (Words Grade Matrix): To align cs_i and cs_j, a value between 0-1 is needed to measure the similarity between any two words $t_{i,k}$ and $t_{j,p}$. This value is denoted as $sim(t_{i,k}, t_{k,p})$, and the word-pair scoring matrix stores all qualifying $sim(t_{i,k}, t_{k,p})$ values for reference during cs_i and cs_j alignment.

(5) **Scoring Rules** G (Grade Rules): When aligning cs_i and cs_j, if the compared words are mismatched or matched with gaps, a penalty of 0.05 points is applied ($G = -0.05$). If the compared words can obtain a 0-1 value by referencing W, this constitutes a similarity match. If the compared words are identical, this constitutes a perfect match with $G = 1$.

(6) **Alignment Score Calculation** S (Final Alignment Scores): After completing the alignment of cs_i and cs_j, each matched word pair receives a score. According to the requirements of different sequence alignment algorithms, the valid scores are accumulated to obtain the final alignment result score S.

The word-pair scoring matrix is the core foundation for applying sequence alignment algorithms to Chinese text similarity research. This matrix provides reference for aligning cs_i and cs_j, thereby better measuring their similarity relationship. The more accurate and standardized the scoring matrix, the better the sequence alignment algorithm performs and the more accurate the alignment results, thus better measuring the similarity between two Chinese texts.

Sequence alignment algorithms work well in bioinformatics because they reference scoring matrices constructed from massive nucleic acid and amino acid statistics. Word2Vec is based on the idea that words with similar contexts have approximate meanings. After training on large corpora, Word2Vec can effectively represent word vectors and quantify word-pair relationships by calculating vector cosine values. This approach closely resembles the bioinformatics methodology for constructing nucleic acid and protein scoring matrices. Therefore, to enable more rational and effective application of sequence alignment

algorithms to Chinese text similarity calculation, this study selected Word2Vec to calculate similarity between Chinese words for constructing the required word-pair scoring matrix.

2.3.2 Global Alignment Algorithm Current research has explored the application of global alignment algorithms to Chinese text similarity calculation. This algorithm aims to analyze the overall similarity relationship between two Chinese sequences, considering the total length of both sequences and comparing all characters to find a solution that maximizes global similarity.

To illustrate the traditional global alignment algorithm, consider the alignment of two Chinese texts: "Research Status and Trends of Knowledge Graphs in Urban Planning Based on GIS" and "Research on Personalized Doctor Recommendations Based on Domestic Medical Knowledge Graphs" . After segmentation, we obtain $cs_i = \{\text{based on, GIS, city, planning, knowledge, graph, research, status, trends}\}$ and $cs_j = \{\text{based on, domestic, medical, knowledge, graph, doctor, personalized, recommendation, research}\}$. Due to the complex and varied combinations of Chinese words, intricate Chinese grammar, and the Chinese expression characteristic of "lighter front, heavier back" where more important content often appears in the latter half, cs_i and cs_j are aligned from tail to head. The alignment process is shown in [Figure 2: see original paper], where scoring follows the scoring rules and the matrix shown in until all words are aligned. Scores are accumulated during alignment, and dynamic programming is used to find the solution with the highest final alignment score as the optimal solution.

Using the word-pair scoring matrix constructed by Word2Vec shown in , we set the scoring condition as $sim(t_{i,k}, t_{k,p}) > 0.65$ based on our trained Word2Vec model. The scoring matrix then contains all word pairs with $sim(t_{i,k}, t_{k,p}) > 0.65$, while all word pairs with $sim(t_{i,k}, t_{k,p}) \leq 0.65$ are uniformly scored as -0.05 according to the scoring rules. When using sequence alignment algorithms to measure similarity between two Chinese sequences, this word-pair scoring matrix provides a reference for comparing word similarities. For example, during alignment, "domestic" and "city" are aligned together with $G = 0.69$ points, while "based on" and "recommendation" are aligned with $G = -0.05$ points.

After alignment completion, the solution with a score of 4.34 shown in [Figure 2: see original paper] is identified as the optimal solution. The alignment path of this optimal solution is then backtracked from head to tail to ensure accuracy, yielding the result shown in . Global alignment produces two Chinese sequences of equal length. Before alignment, cs_i and cs_j have lengths (word counts) of $L_i = 9$ and $L_j = 9$, respectively. After global alignment, gap symbols "-" are inserted into the Chinese sequences due to gap matching (gap symbols are treated as words in alignment), resulting in optimal solution sequence lengths of $L_i = L_j = 12$. Based on the optimal alignment score and sequence length, the similarity between the two sequences is calculated using formula (1): $sim(cs_i, cs_j) = 4.34/12 = 0.362$.

Global alignment is a dynamic programming algorithm with significant redundant calculations. The process of obtaining the optimal score and alignment path involves computations proportional to the size of alignment matrix M ($m \times q$ times), yielding a time complexity of $O(n^2)$.

Despite its effectiveness for texts with minor content differences, global alignment has two major limitations: (1) poor performance when aligning cs_i and cs_j with substantial content differences, and (2) extremely poor performance when aligning sequences with large length differences.

2.3.3 Local Alignment Algorithm Global alignment algorithms aim to find the optimal solution for overall similarity between cs_i and cs_j, while local alignment algorithms seek optimal solutions for local similarity. In local alignment, all scores $S < 0$ are recorded as 0 rather than negative values. After alignment, backtracking returns a subsequence containing the maximum S value rather than complete sequences. Local alignment identifies the solution with the highest S value locally between two Chinese sequences throughout the entire alignment process, but the solution corresponding to S only includes partially aligned words from the two sequences.

Using $cs_i = \{\text{based on, GIS, city, planning, knowledge, graph, research, status, trends}\}$ and $cs_j = \{\text{based on, domestic, medical, knowledge, graph, doctor, personalized, recommendation, research}\}$ for local alignment as an example, [Figure 3: see original paper] shows the alignment matrix where all scores less than 0 are recorded as 0. Aligning from tail to head yields an optimal solution with $S = 3.59$. After alignment, backtracking produces the optimal alignment path shown in . Using formula (2), the similarity is $sim(cs_i, cs_j) = 3.59/(9+9)^2 = 0.399$. Since the local alignment optimal solution only includes partial words from cs_i and cs_j rather than a complete alignment path, the local alignment optimal solution sequence length is the average of the initial lengths of cs_i and cs_j, i.e., $(L_i + L_j)/2 = 9$.

Improved Chinese Sequence Alignment Algorithms

Current research on sequence alignment algorithms primarily focuses on applying global alignment to text similarity calculation, with limited investigation of local alignment algorithms. Moreover, global alignment has certain limitations in text similarity calculation. Therefore, based on existing research, we introduce part-of-speech tagging to better measure word similarity relationships and improve global alignment accuracy, while innovatively applying local alignment algorithms to optimize global alignment.

The specific process of the improved Chinese sequence alignment algorithm is shown in [Figure 4: see original paper]. First, the Chinese sequence set CS to be aligned is constructed. Based on the constructed word-pair scoring matrix and improved scoring rules, an appropriate sequence alignment algorithm is selected to align cs_i and cs_j. Finally, similarity is calculated based on the optimal alignment path. The improved optional sequence alignment algorithms are:

(1) **Global Alignment**: Suitable for aligning cs_i and cs_j with a word count ratio less than 2 (ratio of long to short sequence word counts) and good global similarity.

(2) **Local Alignment**: Suitable for aligning cs_i and cs_j with a word count ratio less than 2 but substantial content differences or low inter-word similarity.

(3) **Multiple Local Alignment**: Suitable for aligning cs_i and cs_j with a word count ratio greater than 2.

When the word count ratio between cs_i and cs_j is less than 2, such sequences can be directly aligned using global alignment. For sequences with good global similarity, high similarity calculation results can be obtained. Then, local alignment is applied to those cs_i and cs_j pairs with low similarity calculation results to identify locally similar sequences, thereby improving similarity calculation results for such cases. When the word count ratio exceeds 2, the longer sequence contains many more words than the short sequence. Using global or local alignment would result in numerous gap matches and poor similarity calculation results. Considering that the longer sequence may contain multiple locations similar to the short sequence, multiple local alignments are performed on such sequences.

3.1 Global Alignment Algorithm Optimization Based on Part-of-Speech Tagging

During global alignment of cs_i and cs_j, word matching scores are provided by the word-pair scoring matrix. For example, "research" and "research" , "service" and "service" both have similarity scores of 1. However, these words may have different usages and parts of speech in different contexts. For instance, both "research" and "service" can function as verb-nouns (vn), verbs (v), and nouns (n). If the aligned words are "research/v" and "research/n" , sim("research/v","research/n") should not be 1. Therefore, when processing Chinese text data with the CRF model, we performed both word segmentation and part-of-speech tagging, fully considering part-of-speech factors to improve scoring rules and enhance the rationality and accuracy of sequence alignment algorithms for Chinese text similarity calculation.

Since part-of-speech tagging results are detailed, covering specific categories like institutions, occupations, and positions, we merged words with the same attributes before alignment based on empirical data. For example, verb-nouns (vn), proper nouns (nx, ng, etc.) were merged into nouns. The improved scoring rules are:

(1) **Perfect Match**: If two perfectly matched words have the same part of speech, score 1 point; if parts of speech differ, deduct 0.05 points ($G = 1 - 0.05 = 0.95$).

(2) **Similarity Match**: For similar matched words with the same

part of speech, score directly from the word-pair scoring matrix ($G = sim(t_{i,k}, t_{k,p})$); if parts of speech differ, deduct 0.05 points ($G = sim(t_{i,k}, t_{k,p}) - 0.05$).

(3) **Mismatch**: For mismatched words with the same part of speech, reward 0.05 points ($G = -0.05 + 0.05 = 0$); if parts of speech differ, $G = -0.05$.

(4) **Gap Match**: Gap matches are uniformly scored as $G = -0.05$.

Using the optimal alignment path for cs_i and cs_j shown in as an example, based on the improved scoring rules, the third word pair is scored as sim("city","domestic") $- 0.05 = 0.64$, and the fourth word pair is scored as $-0.05 + 0.05 = 0$. This standardizes scoring for each matched pair during alignment, further improving global alignment accuracy. Using formula (1), we calculate $sim(cs_i, cs_j) = 4.39/12 = 0.366$.

3.2 Improving Global Alignment Algorithm Using Local Alignment Algorithm

When cs_i and cs_j have substantial content differences, global alignment algorithms are not suitable. Based on the improved scoring rules, we apply local alignment algorithms to recursively solve for local similarity rather than global similarity between two Chinese sequences with large content differences, better measuring and calculating cs_i and cs_j similarity.

Using the optimal alignment path for cs_i and cs_j shown in as an example, all values with $S < 0$ are recorded as 0 in this path. Based on the improved scoring rules, the four mismatched word pairs receive $G = 0.00$ points due to identical parts of speech. Finally, using formula (2), we calculate $sim(cs_i, cs_j) = 3.64/9 = 0.404$.

3.3 Multiple Local Alignment–Improvement Based on Local Alignment Algorithm

When cs_i and cs_j have large sequence length differences, global alignment algorithms are no longer applicable. Based on the improved scoring rules and local alignment algorithm, when the longer sequence contains twice as many words as the short sequence or more, the longer Chinese sequence is first segmented, then locally aligned with the shorter Chinese sequence separately, and finally all alignment results are combined to calculate $sim(cs_i, cs_j)$.

Using $cs_i = \{$scientific knowledge, graph, discipline knowledge, service, application, analysis$\}$ and $cs_j = \{$through, scientific knowledge, graph, humanities, social sciences, discipline, natural science, discipline as an example, cs_i and cs_j contain $L_i = 6$ and $L_j = 19$ words, respectively. First, cs_j is segmented based on cs_i' s word count into three sequences $cs_{j1}, cs_{j2}, cs_{j3}$ containing 6, 6, and 7 words, respectively. Then, cs_i is locally aligned with these three sequences sequentially using the word-pair scoring matrix and improved scoring rules. Finally, combining the

three alignment results (see), similarity is calculated using formula (3): $sim(cs_i, cs_j) = (2.61 + 1.51 + 5.67)/[6 \times 2 + 6 \times 2 + (6 + 7)]/2 = 0.529$.

Experimental Study

This section compares collected Chinese texts and uses representative alignment results to demonstrate and analyze the advantages and accuracy of different sequence alignment algorithms.

4.1 Chinese Text Data Collection and Preprocessing

To test the practical and application value of our methods, we selected online academic resource data as our Chinese text dataset. Given CNKI' s comprehensive literature resources, fast retrieval interface, and accurate batch retrieval capabilities, we exported Chinese literature titles, keywords, and abstracts from CNKI (January 2020 to August 2020) with "knowledge graph" as the search theme. Some retrieved literature titles may not be related to knowledge graphs, but their content might be relevant, so this data was retained. All online academic resource datasets were processed using HanLP' s CRF model for word segmentation and part-of-speech tagging, ultimately yielding an online academic resource Chinese sequence set OARCS containing 747 data entries, with partial data shown in .

For training corpora, we downloaded the latest Chinese Wikipedia corpus XML file from https://dumps.wikimedia.org/zhwiki/latest/, extracted all Chinese text, performed traditional-to-simplified conversion using OpenCC in Python, and again used HanLP' s CRF model for word segmentation to supply subsequent Word2Vec training.

4.2 Construction of Word-Pair Scoring Matrix

The word-pair scoring matrix serves as the core foundation of sequence alignment algorithms, directly affecting final effectiveness and accuracy. Therefore, beyond using Word2Vec to construct the scoring matrix, we also optimized it.

4.2.1 Word2Vec Training and Word-Pair Similarity Calculation We trained Word2Vec' s Skip-Gram model using the processed Wikipedia corpus with the following parameters: word vector dimension $Size = 100$, context window $Windows = 5$, and minimum word frequency $min\_count = 1$. Based on the preprocessed OARCS and trained Word2Vec, we calculated similarity between words. For any two Chinese sequences $cs_i = \{t_{i,1}, t_{i,2}, ..., t_{i,k}, ..., t_{i,m}\}$ and $cs_j = \{t_{j,1}, t_{j,2}, ..., t_{j,p}, ..., t_{j,q}\}$ in OARCS requiring alignment, we used Word2Vec to calculate the cosine similarity $sim(t_{i,k}, t_{j,p})$ between word vectors of any two words $t_{i,k}$ and $t_{j,p}$ as the core foundation for constructing the word-pair scoring matrix.

4.2.2 Optimization of Word-Pair Scoring Matrix Chinese sequence alignment requires appropriate scoring matrix adjustments under different conditions for varying granularity, precision, and accuracy requirements. Similarity values in the scoring matrix range between 0-1.0. As shown in [Figure 5: see original paper], when higher alignment accuracy is required for two texts, the λ value is increased, causing the scoring matrix to retain only word pairs with $sim(t_{i,k}, t_{j,p}) > \lambda$ for alignment scoring, while word pairs with $sim(t_{i,k}, t_{j,p}) < \lambda$ are stored in a non-scoring word library. If λ is adjusted in subsequent alignments, only the scoring matrix and non-scoring word library need to be updated accordingly.

The scoring matrix may contain duplicate $sim(t_{i,k}, t_{j,p})$ values. For example, when both cs_i and cs_j contain the words "research" and "explore" , calculating word-pair similarity yields both sim('research', 'explore') = 0.80 and sim('explore', 'research') = 0.80. To avoid such word pairs occupying additional space in the scoring matrix and causing higher time and space complexity, these duplicate results are removed before placing them in the scoring matrix to improve algorithm efficiency. Finally, for the online academic resource Chinese sequences to be aligned, we constructed the word-pair scoring matrix with $\lambda = 0.7$, retaining all word pairs with $sim(t_{i,k}, t_{j,p}) > 0.7$ in the scoring matrix while storing word pairs with $sim(t_{i,k}, t_{j,p}) \leq 0.7$ in the non-scoring word library, resulting in the scoring matrix shown in .

4.3 Online Academic Resource Chinese Sequence Alignment

Following the previous steps, we obtained the preprocessed OARCS and constructed word-pair scoring matrix. Based on improved scoring rules, Chinese sequences were aligned from tail to head. Partial Chinese sequences for comparison are shown in , where sequences on the left are aligned with those on the right (i.e., sequence with ID cs_1 aligns with sequence with ID cs_2, ID cs_3 with ID cs_4, and so on) until cs_i and cs_j alignment is completed.

4.4 Experimental Results and Evaluation

To highlight the advantages of our research compared with existing studies, we used different sequence alignment algorithms to align and calculate similarity for the Chinese sequences shown in , while setting different word-pair scoring matrix parameters λ to further compare algorithm effectiveness. Results from different algorithms are shown in (where traditional global alignment algorithm similarity calculation results do not reference the constructed scoring matrix or part-of-speech tagging results). Clearly, compared with traditional global alignment algorithms, our method based on Word2Vec-constructed word-pair scoring matrices and part-of-speech tagging results improves overall text similarity calculation.

Examining in detail: For cs_1 vs. cs_2 and cs_3 vs. cs_4, where content and length differences are small (good global similarity and word count ratio less than 2),

global alignment algorithms yield more ideal results. For cs_5 vs. cs_6 and cs_7 vs. cs_8, where content differences are large but length differences are small (good local similarity and word count ratio less than 2), local alignment algorithms outperform global alignment. For cs_9 vs. cs_{10} and cs_{11} vs. cs_{12}, where length differences are substantial (word count ratio greater than 2), multiple local alignment algorithms achieve better results than other algorithms for such sequence similarity calculation.

To more intuitively demonstrate our method' s effectiveness, is visualized as the line chart in [Figure 6: see original paper]. For the first three Chinese sequence pairs, traditional global alignment and improved global alignment ($\lambda = 0.7$) show comparable performance, mainly because when $\lambda = 0.7$, the adjusted scoring matrix contains fewer similar words meeting the condition $sim(t_{i,k}, t_{j,p}) > \lambda$, leading to reduced alignment scores and smaller differences between the two similarity calculation results. When adjusted to $\lambda = 0$, the scoring matrix contains significantly more similar words for scoring reference, and our method' s advantages become more pronounced.

From the empirical process, although Word2Vec used for constructing the scoring matrix can effectively represent training corpora as feature vectors, these word vectors cannot represent the original word order of texts. However, our improved global alignment algorithm and applied local alignment algorithm both strictly follow word order when aligning text similarities. Simultaneously, referencing CRF model part-of-speech tagging results and Word2Vec-constructed word-pair scoring matrices considers both word meanings and improves method effectiveness. Traditional text similarity calculation methods treat words as characters for comparison without considering word meanings and relationships. Our method uses Word2Vec to effectively solve this problem, while the alignment process strictly follows word order, better measuring text similarity relationships.

While improved global alignment algorithms work well for Chinese sequence pairs with good global similarity, they perform poorly for sequences with substantial content or length differences. The local alignment and multiple local alignment algorithms applied in this study effectively address these issues, enabling better application of sequence alignment algorithms to Chinese text similarity calculation.

This study improves global alignment algorithms based on part-of-speech tagging results and constructed word-pair scoring matrices, and applies local alignment algorithms to compensate for global alignment limitations in text similarity calculation: (1) Sequence alignment algorithm effectiveness heavily depends on natural language processing quality in the early research stage. This study uses HanLP' s CRF model for word segmentation, which has good recognition effects for new words, ensuring standardized and accurate Chinese sequence set construction. (2) Part-of-speech judgment for matched word pairs makes alignment scoring more rational, improving sequence alignment algorithm accuracy. The CRF model used in our empirical study integrates HanLP' s core

natural language processing technologies, producing more accurate and detailed part-of-speech tagging results. (3) Word-pair scoring matrix construction is the core foundation for effectively applying sequence alignment algorithms to Chinese text similarity calculation. Although we selected general corpora for Word2Vec training, the latest Wikipedia corpus already possesses sufficient coverage breadth and large scale, making the constructed word-pair scoring matrix highly valuable and effective. (4) We rationally applied local alignment algorithms to better adapt sequence alignment algorithms for text similarity calculation research, laying theoretical and practical foundations for applying these algorithms to semantic mining, text classification and clustering, personalized recommendation, and intelligent retrieval.

Through extensive data testing, the sequence alignment algorithms improved for Chinese "characteristics" based on excellent algorithms and tools in the library and information science field can now be effectively applied to text similarity comparison. Referencing bioinformatics research on sequence alignment algorithms, this study has established a solid foundation for text classification and clustering using sequence alignment algorithms. In the next stage, we will obtain effective Chinese data and, based on the solid theoretical foundation and rich research results of sequence alignment algorithms, attempt to classify and cluster standardized Chinese sequences while combining library and information science research methods and algorithmic tools to further explore deeper relationships and meanings in Chinese texts.

References

[1] MAHMOOD Q, QADIR M A, AFZAL M T. Application of core to computer research papers similarity[J]. IEEE access, 2017, 5: 26124-26134.
[2] PARASCHIV C, DASCALU M, TRAUSAN-MATUS S, et al. Analyzing the semantic relatedness of paper abstracts: an application to the educational research field[C]//International conference on control systems and computer science. Bucharest: IEEE, 2015: 759-764.
[3] HUANG Wenbin, CHE Shangkun. Method system and application analysis of text similarity calculation[J]. Information Theory and Practice, 2019, 42(11): 128-134.
[4] ZHANG Chengzhi. A Chinese string similarity calculation model based on multi-layer features[J]. Journal of the China Society for Scientific and Technical Information, 2005, 24(6): 696-701.
[5] CHEN Erjing, JIANG Enbo. A survey of text similarity calculation methods[J]. Data Analysis and Knowledge Discovery, 2017, 1(6): 1-11.
[6] LI Lin, LI Hui. A text similarity calculation method based on conceptual vector space[J]. Data Analysis and Knowledge Discovery, 2018, 2(5): 48-58.
[7] WANG Chunliu, YANG Yonghui, DENG Fei, et al. A survey of text similarity calculation methods[J]. Information Science, 2019, 37(3): 158-168.
[8] GOMAA W H, FAHMY A A. Short answer grading using string similarity and corpus-based similarity[J]. International journal of advanced computer

science and applications, 2012, 3(11): 114-118.
[9] KADUPITIYA J, RANATHUNGA S, DIAS G. Short sentence similarity calculation using corpus-based and knowledge-based similarity measures[C]//Proceedings of the 26th international conference on computational linguistics. Osaka: The coling 2016 organizing committee, 2016: 44-53.
[10] GOMAA W H, FAHMY A A. A survey of text similarity approaches[J]. International journal of computer applications, 2013, 68(13): 13-18.
[11] BOBADILLA J, ORTEGA F, HERNANDO A, et al. Improving collaborative filtering recommender system results and performance using genetic algorithms[J]. Knowledge-based systems, 2011, 24(8): 1310-1316.
[12] WEN Fengchun, WANG Bangju, XIAO Zhihong. Research status of biological sequence alignment algorithms[J]. Bioinformatics, 2010, 8(1): 66-69.
[13] NEEDLEMAN S B, WUNSCH C D. A general method applicable to the search for similarities in the amino acid sequence of two proteins[J]. Journal of molecular biology, 1970, 48(3): 443-453.
[14] SMITH T F, WATERMAN M S, FITCH W M. Comparative biosequence metrics[J]. Journal of molecular evolution, 1981, 18(1): 38-46.
[15] FENG D F, DOOLITTLE R F. Progressive sequence alignment as a prerequisite to correct phylogenetic trees[J]. Journal of molecular evolution, 1987, 25(4): 351-360.
[16] ALTSCHUL S F, GISH W, MILLER W, et al. Basic local alignment search tool[J]. Journal of molecular biology, 1990, 215(3): 403-410.
[17] EDDY S R. Multiple alignment using hidden markov models[J]. International conference on intelligent systems for molecular biology, 1995(3): 114-120.
[18] THOMPSON J D, HIGGINS D G, GIBSON T J. Clustal w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice[J]. Nucleic acids research, 1994, 22(22): 4673-4680.
[19] NOTREDAME C, HIGGINS. T-coffee: a novel method for fast and accurate multiple sequence alignment[J]. Journal of molecular biology, 2000, 302(1): 205-217.
[20] LASSMANN T. Kalign3: multiple sequence alignment of large datasets[J]. Bioinformatics, 2019, 36(6): 1928-1929.
[21] ZHANG C X, ZHENG W, MORTUZA S M, et al. Deepmsa: constructing deep multiple sequence alignment to improve contact prediction and fold-recognition for distant-homology proteins[J]. Bioinformatics, 2019, 36(7): 2105-2112.
[22] LU R J, ZHAO X, LI J, et al. Genomic characterisation and epidemiology of 2019 novel coronavirus: implications for virus origins and receptor binding[J]. The lancet, 2020, 395: 565-574.
[23] XU Shuo, ZHU Lijun, QIAO Xiaodong, et al. A new method for calculating semantic similarity of Chinese terms based on dual-sequence alignment[J]. Journal of the China Society for Scientific and Technical Information, 2010, 29(4): 701-708.
[24] WANG Ting, XU Tiansheng, JI Fujun. Large-scale Chinese linked data model based on data field and global sequence alignment[J]. Chinese Journal of

Information Processing, 2016, 30(3): 204-212.
[25] TIAN Jiule, ZHAO Wei. Word similarity calculation method based on synonym forest[J]. Journal of Jilin University (Information Science Edition), 2010, 28(6): 602-608.
[26] XIONG Huixiang, ZHAO Dengpeng, LU Chenfan. Research on Chinese sequence alignment based on word vector model[J]. Library and Information Service, 2020, 65(10): 86-98.
[27] SUTTON B C, MCCALLUM A. An introduction to conditional random fields[J]. Foundations & trends in machine learning, 2010, 4(4): 267-373.
[28] BUCHHOLZ S, MARSI E. Conll-x shared task on multilingual dependency parsing[C]//Tenth conference on computational natural language learning. New York: Association for computational linguistics, 2006.
[29] JOULIN A, GRAVE E, BOJANOWSKI P, et al. Bag of tricks for efficient text classification[C]//Proceedings of the 15th conference of the European chapter of the Association for Computational Linguistics, 2017: 427-431.
[30] DOZAT T, MANNING C D. Deep biaffine attention for neural dependency parsing[C]//The 5th international conference on learning representations. Puerto Rico: ICLR, 2016: 1-8.
[31] DEVLIN J, CHANG M W, LEE K, et al. Bert: pre-training of deep bidirectional transformers for language understanding[EB/OL]/[2021-01-30]. http://arXiv:1810.04805.
[32] MIKOLOV T, SUTSKEVER I, CHEN K, et al. Distributed representations of words and phrases and their compositionality[J]. Advances in neural information processing systems, 2013, 26: 3111-3119.
[33] GUO Sicheng, LI Gang, ZHOU Huayang. Research on interoperability of medical knowledge organization systems based on Word2Vec: taking semantic mapping between vocabularies as an example[J]. Information Theory and Practice, 2019, 42(9): 160-165.

Author Contributions:
Zhao Dengpeng: Algorithm design and paper writing;
Xiong Huixiang: Research direction and methodology proposal;
Tian Fengshou: Technical support and guidance;
Li Xinran: Paper proofreading and revision.

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